VERIFICATION OF TRANSLATION

I, the below named translator, hereby declare:

that my name and my post office address are as stated below; and

that I am knowledgeable in the English and Japanese languages and that I believe the following is a true and complete translation into the English language of Japanese Patent Application No. 11-201929 filed in the Japanese Patent Office on the 15th day of July 1999 for Letters Patent.

Signed this 27th day of June 2005

Matsuo Kimata
Full Name of Translator

Matsuo Kimata
Signature of Translator

Post Office Address
Twin-Cross 1001, Akashi-cho 1-3
Chuo-ku, Tokyo

[TITLE OF THE DOCUMENT]
Specification

[TITLE OF THE INVENTION]
Circuit and Method for Designing Tree-structured Communication Channel and Computer-readable Storage Medium

[PATENT CLAIMS]
[Claim 1]
A circuit for designing a tree-structured communication path for a communications network which comprises an egress node from which communication traffic is delivered, a plurality of ingress nodes to which said communication traffic is entered, and a transit node for repeating the communication traffic from said ingress node to said egress node, and a plurality of communication links for interconnecting said ingress, transit and egress nodes, said designing circuit determining a tree-structured communication path that accommodates all possible paths between a given ingress node and a given egress node, characterized by:

optimizing threshold generating means for generating an object function that minimizes the number of candidate tree graphs for accommodating said paths;

tree defining equation generating means for generating a constraint equation for causing all of said candidate tree graphs to form a tree;

path accommodating equation generating means for generating a constraint equation for accommodating said communication paths in one of said candidate tree graphs;

tree utilization check equation generating means for defining a constraint equation for determining whether each of said candidate tree graphs is used to accommodate said communication paths; and

optimizing means for solving a compound integer programming problem formed by said objective function, and said constraint equations to obtain a plurality of trees in which said communication paths can be accommodated.

[Claim 2]
A circuit for designing a tree-structured communication path for a communications network which comprises an egress node from which communication traffic is delivered, a plurality of ingress nodes to which said communication traffic is entered, and a transit node for repeating the communication traffic from said ingress node to said egress node, and a plurality of communication links for interconnecting said ingress, transit and

egress nodes, said designing circuit determining a tree-structured communication path that accommodates all paths when existing paths are given and a path is given between an ingress node and an egress node, characterized by:

existing path accommodating means for accommodating said existing path in an existing tree-structured communication path;

optimizing threshold generating means for generating an objective function that minimizes the number of candidate tree graphs that accommodate ones of said communication paths which cannot be accommodated in said existing tree;

tree defining equation generating means for defining a constraint equation for causing all of said candidate tree graphs to form a tree;

path accommodating equation generating means for defining a constraint equation for accommodating said ones of communication paths in one of said candidate tree graphs;

tree utilization check equation generating means for defining a constraint equation for determining whether each of said candidate tree graphs is used to accommodate at least one of said communication paths; and

optimizing means for solving a compound integer programming problem formed by said objective function, and said constraint equations to obtain a plurality of trees in which said ones of communication paths can be accommodated.

[Claim 3]

A circuit for designing a tree-structured communication path for a communications network which comprises an egress node from which communication traffic is delivered, a plurality of ingress nodes to which said communication traffic is entered, and a transit node for repeating the communication traffic from said ingress node to said egress node, and a plurality of communication links for interconnecting said ingress, transit and egress nodes, said designing circuit determining a tree-structured communication path that accommodates all existing paths when a path is given between an ingress node and an egress node, characterized by:

tree defining means for defining a constraint equation for causing all candidate tree graphs to form a tree;

path accommodating means for defining a constraint equation for accommodating said communication paths in one of said candidate tree graphs;

artificial variable embedding means for embedding non-negative artificial variables into said constraint equations defined by said tree defining means and said path accommodating means;

realizability decision threshold generating means for defining an

objective function for minimizing a total number of said non-negative artificial variables; and

optimizing means for solving a compound integer programming problem formed by said objective function, and said constraint equations defined by said tree defining means and said path accommodating means to obtain a plurality of trees in which said communication paths can be accommodated.

[Claim 4]

A circuit for designing a tree-structured communication path for a communications network which comprises an egress node from which communication traffic is delivered, a plurality of ingress nodes to which said communication traffic is entered, and a transit node for repeating the communication traffic from said ingress node to said egress node, and a plurality of communication links for interconnecting said ingress, transit and egress nodes, said designing circuit determining a tree-structured communication path that accommodates all existing paths when a path is given between an ingress node and an egress node, characterized by:

existing path accommodating means for accommodating said given tree in said existing tree;

path accommodating equation generating means for defining a constraint equation for accommodating ones of said communication paths which cannot be accommodated in said existing tree in one of said candidate tree graphs;

tree defining equation generating means for defining a constraint equation for causing all of said candidate tree graphs to form a tree;

artificial variable embedding means for embedding non-negative artificial variables into said constraint equations formed by said path accommodating means and said tree defining means;

realizability decision threshold generating means for defining an objective function for minimizing a total number of said non-negative artificial variables; and

optimizing means for solving a compound integer programming problem formed by said objective function, and said constraint equations formed by said path accommodating means and said tree defining equation generating equation generating means to obtain a plurality of trees in which said ones of communication paths can be accommodated.

[Claim 5]

A method of designing a tree-structured communication path for a communications network which comprises an egress node from which communication traffic is delivered, a plurality of ingress nodes to which said communication traffic is entered, and a transit node for repeating the

communication traffic from said ingress node to said egress node, and a plurality of communication links for interconnecting said ingress, transit and egress nodes, said designing circuit determining a tree-structured communication path that accommodates all possible paths between a given ingress node and a given egress node, characterized by the steps of:

generating an object function that minimizes the number of candidate tree graphs for accommodating said paths;

generating a constraint equation for causing all of said candidate tree graphs to form a tree;

generating a constraint equation for accommodating said communication paths in one of said candidate tree graphs;

defining a constraint equation for determining whether each of said candidate tree graphs is used to accommodate said communication paths; and

solving a compound integer programming problem formed by said objective function, and said constraint equations to obtain a plurality of trees in which said communication paths can be accommodated.

[Claim 6]

A method of designing a tree-structured communication path for a communications network which comprises an egress node from which communication traffic is delivered, a plurality of ingress nodes to which said communication traffic is entered, and a transit node for repeating the communication traffic from said ingress node to said egress node, and a plurality of communication links for interconnecting said ingress, transit and egress nodes, said designing circuit determining a tree-structured communication path that accommodates all paths when existing paths are given and a path is given between an ingress node and an egress node, characterized by the steps of:

accommodating said existing path in an existing tree-structured communication path;

generating an objective function that minimizes the number of candidate tree graphs that accommodate ones of said communication paths which cannot be accommodated in said existing tree;

defining a constraint equation for causing all of said candidate tree graphs to form a tree;

defining a constraint equation for accommodating said ones of communication paths in one of said candidate tree graphs;

defining a constraint equation for determining whether each of said candidate tree graphs is used to accommodate at least one of said communication paths; and

solving a compound integer programming problem formed by said

objective function, and said constraint equations to obtain a plurality of trees in which said ones of communication paths can be accommodated.

[Claim 7]

A method of designing a tree-structured communication path for a communications network which comprises an egress node from which communication traffic is delivered, a plurality of ingress nodes to which said communication traffic is entered, and a transit node for repeating the communication traffic from said ingress node to said egress node, and a plurality of communication links for interconnecting said ingress, transit and egress nodes, said designing circuit determining a tree-structured communication path that accommodates all existing paths when a path is given between an ingress node and an egress node, characterized by the steps of:

defining a constraint equation for causing all candidate tree graphs to form a tree;

defining a constraint equation for accommodating said communication paths in one of said candidate tree graphs;

embedding non-negative artificial variables into said constraint equations defined by said tree defining means and said path accommodating means;

defining an objective function for minimizing a total number of said non-negative artificial variables; and

solving a compound integer programming problem formed by said objective function, and said constraint equations to obtain a plurality of trees in which said communication paths can be accommodated.

[Claim 8]

A method of designing a tree-structured communication path for a communications network which comprises an egress node from which communication traffic is delivered, a plurality of ingress nodes to which said communication traffic is entered, and a transit node for repeating the communication traffic from said ingress node to said egress node, and a plurality of communication links for interconnecting said ingress, transit and egress nodes, said designing circuit determining a tree-structured communication path that accommodates all existing paths when a path is given between an ingress node and an egress node, characterized by:

accommodating said given tree in said existing tree;

defining a constraint equation for accommodating ones of said communication paths which cannot be accommodated in said existing tree in one of said candidate tree graphs;

defining a constraint equation for causing all of said candidate tree graphs to form a tree;

embedding non-negative artificial variables into said constraint equations formed by said path accommodating means and said tree defining means;

defining an objective function for minimizing a total number of said non-negative artificial variables; and

solving a compound integer programming problem formed by said objective function, and said constraint equations formed by said path accommodating means and said tree defining means to obtain a plurality of trees in which said ones of communication paths can be accommodated.

[Claim 9]

A computer-readable storage medium recording a program of designing a tree-structured communication path for a communications network is determined which comprises an egress node from which communication traffic is delivered, a plurality of ingress nodes to which said communication traffic is entered, and a transit node for repeating the communication traffic from said ingress node to said egress node, and a plurality of communication links for interconnecting said ingress, transit and egress nodes, said designing circuit determining a tree-structured communication path that accommodates all possible paths between a given ingress node and a given egress node, wherein said program is characterized by the steps of:

generating an object function that minimizes the number of candidate tree graphs for accommodating said paths;

generating a constraint equation for causing all of said candidate tree graphs to form a tree;

generating a constraint equation for accommodating said communication paths in one of said candidate tree graphs;

defining a constraint equation for determining whether each of said candidate tree graphs is used to accommodate said communication paths; and

solving a compound integer programming problem formed by said objective function, and said constraint equations to obtain a plurality of trees in which said communication paths can be accommodated.

[Claim 10]

A computer-readable storage medium recording a program of designing a tree-structured communication path for a communications network is determined which comprises an egress node from which communication traffic is delivered, a plurality of ingress nodes to which said communication traffic is entered, and a transit node for repeating the communication traffic from said ingress node to said egress node, and a plurality of communication links for interconnecting said ingress, transit and

egress nodes, said designing circuit determining a tree-structured communication path that accommodates all paths when existing paths are given and a path is given between an ingress node and an egress node, wherein said program is characterized by the steps of:

accommodating said existing path in an existing tree-structured communication path;

generating an objective function that minimizes the number of candidate tree graphs that accommodate ones of said communication paths which cannot be accommodated in said existing tree;

defining a constraint equation for causing all of said candidate tree graphs to form a tree;

defining a constraint equation for accommodating said ones of communication paths in one of said candidate tree graphs;

defining a constraint equation for determining whether each of said candidate tree graphs is used to accommodate at least one of said communication paths; and

solving a compound integer programming problem formed by said objective function, and said constraint equations to obtain a plurality of trees in which said ones of communication paths can be accommodated.

[Claim 11]

A computer-readable storage medium containing a program of designing a tree-structured communication path for a communications network is determined which comprises an egress node from which communication traffic is delivered, a plurality of ingress nodes to which said communication traffic is entered, and a transit node for repeating the communication traffic from said ingress node to said egress node, and a plurality of communication links for interconnecting said ingress, transit and egress nodes, said designing circuit determining a tree-structured communication path that accommodates all existing paths when a path is given between an ingress node and an egress node, wherein said program is characterized by the steps of:

defining a constraint equation for causing all candidate tree graphs to form a tree;

defining a constraint equation for accommodating said communication paths in one of said candidate tree graphs;

embedding non-negative artificial variables into said constraint equations defined by said tree defining means and said path accommodating means;

defining an objective function for minimizing a total number of said non-negative artificial variables; and

solving a compound integer programming problem formed by said

objective function, and said constraint equations to obtain a plurality of trees in which said communication paths can be accommodated.

[Claim 12]

A computer-readable storage medium of designing a tree-structured communication path for a communications network is determined which comprises an egress node from which communication traffic is delivered, a plurality of ingress nodes to which said communication traffic is entered, and a transit node for repeating the communication traffic from said ingress node to said egress node, and a plurality of communication links for interconnecting said ingress, transit and egress nodes, said designing circuit determining a tree-structured communication path that accommodates all existing paths when a path is given between an ingress node and an egress node, wherein said program is characterized by the steps of:

accommodating said given tree in said existing tree;

defining a constraint equation for accommodating ones of said communication paths which cannot be accommodated in said existing tree in one of said candidate tree graphs;

defining a constraint equation for causing all of said candidate tree graphs to form a tree;

embedding non-negative artificial variables into said constraint equations formed by said path accommodating means and said tree defining means;

defining an objective function for minimizing a total number of said non-negative artificial variables; and

solving a compound integer programming problem formed by said objective function, and said constraint equations formed by said path accommodating means and said tree defining means to obtain a plurality of trees in which said ones of communication paths can be accommodated.

[DETAILED DESCRIPTION OF THE INVENTION]

[0001]

[Technological Field of the Invention]

The present invention relates to a design technique for a tree-structured communication links and more specifically to a design technique for designing a tree-structure communication path in which the number of trees is minimized when all given paths are accommodated between an ingress node and an egress node.

[0002]

[Prior Art]

Communication by establishing a path in a communications network between an ingress node and an egress and sending a packet over the

established path is a well-known technique available in the form of the virtual channel (VC) or virtual path (VP) of asynchronous transfer mode (ATM) system, and the label switched path (LSP) of multiprotocol label switching (MPLS) (A framework for Multiprotocol Label Switching Architecture, http//www.ietf.org/internet-drafts/draft-ietf-mpls-framework-02.txt).

[0003]

This mode of communication needs at least one path to be established between an ingress node where the demand is generated and an egress node. In order for the network to find an alternate path for a failure in a network node or a network link, it is necessary to define a set of paths which do not share the same nodes and links. In addition, multiple paths need to be established between an ingress node and an egress node to avoid concentration of traffic on a particular node or link.

[0004]

However, to establish multiple paths between all ingress nodes and all egress nodes a great number of VPIs and VCIs would be required in the case of ATM and a great number of labels would also be required in the case of MPLS systems.

[0005]

In the case of ATM systems, a technique known as VP merge or VC merge is used, which involves the use of the same VPI and VCI for a multipoint-to-point communication between a plurality of ingress nodes to a single egress node ("A Framework for Multiprotocol Label Switching Architecture", http://www. Ietf. org/ internet-drafts/draft-ietf-mpls-framework-02.txt). In the case of MPLS, a similar technique is used for a multipoint-to-point communication in which the same label is used between a plurality of ingress nodes and a single egress node ("IP Navigator MPLS Executive Overview", Ascend White Paper, http://www.ascend.com/docs/ Techdocs/ipnavwp.pdf).

[0006]

Using the same VPI/VCI or label for a multipoint-to-point communication indicates that the transmission in the opposite direction is a transmission through a tree structure from a root node. It is necessary to provide a tree structure that requires a small number of VPIs and VCIs while meeting the requirements that there be at least one pair of paths that do not share the same link or node and that there be a plurality of paths to avoid traffic concentration.

[0007]

Another method of forming a network of tree structure involves the use of the Dijkstra principle as described in Optimizing Handbook, Iri, Konno,

Tone, Asakura Book Company. The Dijkstra method is one that makes a search for a shortest distance from a source point to every other apex of the tree. By starting the search for a shortest path from the egress node to every ingress node, a network of tree structure can be built.

[0008]

Another method of building a tree structured network is the minimum spanning tree method (described in Graph Theory, Iri, Shirakawa, Kajiya, Shinoda, Corona Book Company). The minimum spanning tree is a tree where the total sum of the weights of its metrics is at a minimum. The Krustal method can be used to build a minimum spanning tree.

[0009]

[Problems Solved by the Invention]

The prior art method builds only one tree structured network given a network topology. For this reason, there is only one path between an ingress node and an egress node. It is thus impossible to provide a pair of paths not sharing the same node or link as an alternate path for routing the traffic around a failed point and further impossible to provide a plurality of paths for balancing traffic loads.

[0010]

[Object of the Invention]

The present invention has the object of providing a technique for designing a tree structured communication medium that can accommodate given paths with a minimum number of trees. More specifically, the present invention provides a pair of paths not sharing the same node or link as an alternate path for routing the traffic around a failed point and further impossible to provide a plurality of paths for balancing traffic loads, while the number of trees is kept at a minimum.

[0011]

Another object of the present invention is to provide a technique for designing a tree structured communication medium that can accommodate given paths. More specifically, the present invention provides a pair of paths not sharing the same node or link as an alternate path for routing the traffic around a failed point and further impossible to provide a plurality of paths for balancing traffic loads.

[0012]

[Means for Solving the Problems]

The design circuit of the present invention designs a first tree-structured path by setting a compound integer programming problem that can accommodate a communication path of minimum number of trees between a given ingress node and a given egress node, and solving the

problem. More specifically, there are provided optimizing threshold generating means (101, Fig. 1) for generating an object function that minimizes the number of tree graphs for accommodating said paths, tree defining equation generating means (102, Fig. 1), path accommodating equation generating means (103, Fig. 1), tree utilization check equation generating means (104, Fig. 1) and optimizing means (105, Fig. 1) for solving a compound integer programming problem formed by these means.

[0013]

The design circuit of the present invention designs a second tree-structured path by accommodating an existing tree-structured communication path between a given ingress node and a given egress node, and then setting a compound integer programming problem with respect to those paths where said existing path cannot be accommodated for accommodating a tree-structured communication path of a minimum number of trees and solves the compound integer programming problem. More specifically, there are provided existing path accommodating means (501, Fig. 5) for accommodating said existing path in an existing tree-structured communication path, optimizing threshold generating means (502, Fig. 5) for generating a compound integer programming problem that minimizes the number of trees that accommodate ones of said communication paths which cannot be accommodated in said existing tree, tree defining equation generating means (503, Fig. 5), path accommodating means (504, Fig. 5), tree utilization check equation generating means (505, Fig. 5), and optimizing means (506, Fig. 5) for solving the compound integer programming problem.

[0014]

The design circuit of the present invention designs a third tree-structured tree-structured path by accommodating an existing tree-structured communication path between a given ingress node and a given egress node, and then setting a compound integer programming problem with respect to those paths where said existing path cannot be accommodated for accommodating a tree-structured communication path of a minimum number of trees and solves the compound integer programming problem. More specifically, there are provided realizability decision threshold generating means (801, Fig. 8) for generating this compound integer programming problem, tree defining threshold generating means (802, Fig. 8), path accommodating means (803, Fig. 8), artificial variable embedding means (804, Fig. 8) and optimizing means (805, Fig. 8) for solving the compound integer programming problem.

[0015]

The design circuit of the present invention designs a fourth tree-structured path by accommodating an existing tree-structured

communication path between a given ingress node and a given egress node, and then setting a compound integer programming problem with respect to those paths where said existing path cannot be accommodated for accommodating a tree-structured communication path of a minimum number of trees and solves the compound integer programming problem. More specifically, there are provided existing path accommodating means (1101, Fig. 11) for accommodating said existing path in an existing tree-structured communication path, realizability decision threshold generating means (1102, Fig. 11) for generating this compound integer programming problem, tree defining equation generating means (1103, Fig. 11), path accommodating means (1104, Fig. 11), artificial variable embedding means (1105, Fig. 11) and optimizing means (1106, Fig. 11) for solving the compound integer programming problem.

[0016]

A compound integer programming problem is generated for accommodating all possible paths in a minimum number of tree structured communication path by using the optimizing threshold generating means, the tree defining equation generating means, the path accommodating equation generating means, and the tree utilization check equation generating means and the optimizing means is used to solve the compound integer programming problem.

[0017]

In addition, a compound integer programming problem is generated for accommodating all possible paths in a minimum number of tree structured communication path by using the realizability decision threshold generating means, the tree defining equation generating means, the path accommodating equation generating means, and the artificial variable embedding means and the optimizing means is used to solve the compound integer programming problem.

[0018]

Furthermore, the existing path tree accommodating means is used accommodate the path between the ingress node and the egress node which are given to existing trees.

[0019]

[Mode of Implementation of the Invention]

The following is a description of a mode of implementation of the present invention with threshold to the accompanying drawings.

[0020]

Fig. 2 illustrates one example of the network on which the design technique of the present invention is applied. In the present invention, the network is represented by a directed graph. If the present invention is used

for designing a network represented by a non-directed graph, the network is treated link-by-link basis.

[0021]

The node of a communications network to which user traffic is entered is called an ingress node and the node through which the traffic exits from the network is called an egress node, and a node that can function both as an ingress node and an egress node is called a terminal node. In Fig. 2, nodes e_1 through node e_{10} are terminal nodes. Nodes that neither act as ingress or egress nodes are called a transit node. In Fig. 2, nodes c_1 through c_5 are transit nodes. Furthermore, a directed link exists between any pair of nodes. In Fig. 2, there is one non-directed line. This line means that a bidirectional link exists in the network. A link between nodes e_5 and node c_3, for example, is designated as link (e_5, c_3). In Fig. 2, a tree is represented by a set of arrow-headed thick lines of the same attribute. In this example, only those thick lines necessary for disclosure are illustrated. In Fig. 2, three types of tree t_1, t_2 and t_3 are shown. A communication path is represented by a concatenation of node indicators. For example, a path that extends through nodes e_7, c_5, c_1 and e_1 is designated as $e_7 - c_5 - c_1 - e_1$.

[0022]

The following is a description of the network design method of the present invention, using node e_1 as an egress node, for establishing two paths $e_7 - c_5 - c_1 - e_1$ and $e_7 - c_4 - c_3 - c_2 - e_1$ from node e_7 to node e_1. It is seen that these paths can be accommodated in trees t_1 and t_2. Further, if paths $e_5 - c_4 - c_3 - e_1$ and $e_5 - c_3 - c_4 - c_1 - e_1$ are established from node e_5 to node e_1, it is seen that these paths can be accommodated in trees t_1 and t_2 as in the first example. If paths $e_3 - c_2 - e_1$ and $e_3 - c_3 - c_4 - c_1 - e_1$ are established from node e_3 to node e_1, it is seen that the path $e_3 - c_2 - e_1$ can be accommodated in tree t_2 but the path $e_3 - c_3 - c_4 - c_1 - e_1$ can not be accommodated in either tree t_1 or t_2. For accommodating the path $e_3 - c_3 - c_4 - c_1 - e_1$, a tree t_3 is generated. In the last example, three trees are used.

[0023]

The following is a definition of the terms and symbols used in the present invention.

[0024]

First comes the definition of a candidate tree graph. In the communications network of interest, a candidate tree graph is represented by an egress node, a plurality of ingress nodes and all transit nodes and links interconnecting these nodes, and those links that are reachable from any of the ingress nodes to the egress node. In the present invention, this tree candidate graph accommodates given paths and a compound integer programming

problem is set and solved.

[0025]

The following is a definition of symbols of sets and elements.

[0026]

e : egress node;

T_e : a set of candidate tree graphs at the egress node e;

N^{core}: a set of transit nodes;

N^{edge}: a set of terminal nodes;

L^{c-c}: a set of links interconnecting transit nodes, where each element is represented by (l, m), where l is a source node and m is a destination node.

L^{e-c}: a set of links interconnecting terminal nodes and transit nodes, where each element is represented by (l, m), where l is a source node and m is a destination node. One of the elements (l, m) is a transit node and the other is a terminal node;

$P_{(i, e)}$: a set of paths between an ingress node i and an egress node e, where each element is represented by $p_{(i, e)}$; and

$L^{p(i, e)}$: a set of links used on a path p(i, e).

[0027]

The following is a definition of variables.

[0028]

r^{te}: a 0-1 variable that equals 1 when the candidate tree graph t_e of egress node e is used to accommodate a path; otherwise 0;

$h^{te}{}_{(l,m)}$: a 0-1 variable that equals 1 when candidate tree graph t_e uses link (l, m); otherwise 0;

$f^{te}{}_{(l, m)}$: a real number representing a volume of traffic that flows through link (l, m) at the candidate tree graph t_e, where the flow corresponds to what is defined in the literature "Optimizing Handbook", Iri, Konno, and Tone, the Asakura Book Company; and

$\delta^{te}{}_{p(i, e)}$: a 0-1 variable that is equal to 1 when the candidate tree graph t_e contains a pth $p_{(i, e)}$; otherwise 0.

[0029]

Finally, constants are defined as follows.

[0030]

$o_{(l, e)}$: a constant that assumes 1 if a link (l, e) exists that connect a transit node l and an egress node e; otherwise 0.

M : a large arbitrary value.

[0031]

The following is a description of a first embodiment of the present invention.

[0032]

Fig. 1 is a block diagram of a design circuit for designing a tree structured transmission channel according to a first mode of implementation of the present invention. The design circuit comprises a computer 100 including an optimizing threshold generating means 101, a tree defining equation generating means 102, a path accommodating equation generating means 103, a tree utilization check equation generating means 104 and an optimizing means 105. A keyboard 106, a display unit and output device 107 and a storage medium 108 are connected to the computer 100.

[0033]

Storage medium 108 may be a floppy disk, a read-only memory or any other types of store. Storage medium 108 is used to store a program for enabling the computer 100 to function as a design circuit for designing a tree structured communication channel. This program is read by the computer 100 to control its function so that the optimizing threshold generating means 101, the tree defining equation generating means 102, the path accommodating equation generating means 103, the tree utilization check equation generating means 104 and the optimizing means 105 are implemented.

[0034]

Fig. 3 is a flowchart of one example of the operation of the optimizing threshold generating means 101, the tree defining equation generating means 102, the path accommodating equation generating means 103, the tree utilization check equation generating means 104 and the optimizing means 105 according to the first mode of implementation of the present invention.

[0035]

First, there are given, as input data to the computer 100 by using the input device 106, a network topology comprising terminal nodes, transit nodes and interconnecting links, the node identifier of an egress node, a set of paths between ingress nodes and the egress node, and the number of candidate tree graphs. If the number of candidate tree graphs is relatively small, it is possible that all the given paths cannot be accommodated. The number of candidate tree graphs may be determined repeatedly on a trial-and-error basis.

[0036]

Optimizing threshold generating means 101 uses the input data to generate an object function according to Equation (1) given below (see step 301, Fig. 3).

[0037]

$$\text{Minimize} \sum_{t_e \in T_e} r^{t_e} \qquad (1)$$

[0038]

Equation (1) is the object function that minimizes the number of candidate tree graphs which will be used to accommodate the given paths.

[0039]

Next, the tree defining equation generating means 102 generates a first constraint equation necessary to enable the candidate tree graphs to form a tree, i.e., the constraint that concatenate the candidate tree graphs, and a second constraint equation necessary to enable the candidate tree graphs to form links whose number is equal to (the number of nodes minus one) (see step 302, Fig. 3).

[0040]

First, the concatenating constraint equation is generated. In order for the candidate tree graphs to form a concatenation, a network flow is formed by setting the egress node as an inlet port and setting the ingress and transit nodes as outlet ports. The following is a description of constraint equations (2), (3) and (4).

[0041]

$$\sum_{\{m:(l,m)\in L^{e-c}\}} f_{(l,m)}^{t_e} = 1 \quad (\forall t_e \in T_e, \forall \ell \in N^{edge} \setminus \{e\}) \qquad (2)$$

[0042]

$$\sum_{\{m:(l,m)\in L^{c-c}\}} f_{(l,m)}^{t_e} - \sum_{\{m:(l,m)\in L^{c-c}\}} f_{(m,l)}^{t_e} + o(l,e)\ f_{(l,e)}^{t_e}$$

$$- \sum_{\{m:(m,l)\in L^{e-c}\}} f_{(m,l)}^{t_e} = 1 \quad (\forall t_e \in T_e, \ell \in N^{core}) \qquad (3)$$

[0043]

$$\sum_{\{l:(l,e)\in L^{e-c}\}} f_{(l,e)}^{t_e} = \left|N^{edge}\right| + \left|N^{core}\right| - 1 \quad (\forall t_e \in T) \qquad (4)$$

[0044]

Constraint equation (2) indicates that ingress nodes are functioning as outlet nodes where traffic flows out. Constraint equation (3) indicates that transit nodes are functioning as outlet ports where traffic flows out. Constraint equation (4) indicates a total of traffic volumes $|N^{edge}| + |N^{core}| - 1$ carried by the egress node functioning as an inlet port where traffic flows in

and by the ingress and transit nodes functioning as outlet ports.

[0045]

Next, a constraint equation (5) for setting the number of links equal to (the number of nodes minus 1) is created.

[0046]

$$\sum_{\{(l,m)\in L^{c-c}\}} h^{t_e}_{(l,m)} - \sum_{\{l:(l,e)\in E^{e-c}\}} h^{t_e}_{(l,e)}$$

$$+ \sum_{l\in N^{edge}\setminus\{e\}} \sum_{\{m:(l,m)\in E^{e-c}\}} h^{t_e}_{(l,e)} = \left|N^{core}\right| + \left|N^{edge}\right| - 1 \qquad (5)$$

[0047]

While the constraint equations (2) to (4) use the variable $f^{te}_{(l,m)}$ for concatenation, the constraint equation (5) uses the variable $h^{te}_{(l,m)}$. It is thus necessary to establish correlation between these variables according to the following manner.

[0048]

$$Mh^{t_e}_{(l,m)} \geq f^{t_e}_{(l,m)} \quad (\forall t_e \in T_e, \forall (l,m) \in L^{c-c}) \qquad (6)$$

[0049]

$$Mh^{t_e}_{(l,e)} \geq f^{t_e}_{(l,e)} \quad (\forall t_e \in T_e, \forall (l,e) \in L^{e-c}) \qquad (7)$$

[0050]

$$Mh^{t_e}_{(l,m)} \geq f^{t_e}_{(l,m)} \quad (\forall t_e \in T_e, \forall (l,m) \in L^{e-c}, \forall l \in N^{edge}\setminus\{e\}) \qquad (8)$$

[0051]

Constraint equation (6) correlates the variables associated with the links interconnecting the transit nodes. Constraint equation (7) correlates the variables associated with the links connecting the transit nodes to the egress node, and the constraint equation (8) correlates the variables associated with the links connecting the ingress nodes to the transit nodes.

[0052]

Note that in the constraint equations (2) and (3), the fourth term uses the variables $f^{te}_{(l,m)}$ and $f^{te}_{(m,l)}$, these variables can be replaced with $h^{te}_{(l,m)}$ and $h^{te}_{(m,l)}$, respectively, to delete equation (8). Constraint equation (5) can be rewritten as follows:

[0053]

$$\sum_{\{(l,m)\in L^{c-c}\}} h^{t_e}_{(l,m)} - \sum_{\{l:(l,e)\in E^{e-c}\}} h^{t_e}_{(l,e)} = \left|N^{core}\right| \quad (\forall t_e \in T_e) \qquad (9)$$

[0054]

Next, the path accommodating equation generating means 103 defines the following constraint equations (10) and (11) for accommodating the given paths (see step 303, Fig. 3).

[0055]

$$\sum_{(l,m)\in\left\{L^{p(i,e)}\cap L^{c-c}\right\}} h_{(l,m)}^{t_e} + \sum_{(l,m)\in\left\{L^{p(i,e)}\cap L^{e-c}\right\}} h_{(l,m)}^{t_e} \geq \left|L^{p(i,e)}\right| \delta_{p(i,e)}^{t_e} \quad \left(p(i,e) \in P_{(i,e)}, i \in N^{edge} \setminus \{e\}, t_e \in T_e\right) \tag{10}$$

[0056]

$$\sum_{t_e \in T_e} \delta_{p(i,e)}^{t_e} \geq 1 \quad \left(\forall p(i,e) \in P_{(i,e)}, \forall_i \in N \setminus \{e\}\right) \tag{11}$$

[0057]

Constraint equation (10) means that if the addition of the link-associated variable $h^{te}{}_{(l,m)}$ used by the path $p_{(i,\ e,)}$ equals the hop count number of p(i, e), the paths are accommodated in the candidate tree graph t_e. Constraint equation (11) indicates that all the given paths are accommodated in either of the candidate tree graphs.

[0058]

Next, the tree utilization check equation generating means 104 defines the following constraint equation (12) for making a decision in which of the candidate tree graphs the given path is accommodated (see step 304, Fig 3).

[0059]

$$\sum_{i\in N^{edge} \setminus \{e\}} \sum_{P_{(i,e)} \in p(i,e)} \delta_{p(i,e)}^{t_e} \leq M\, r^{t_e} \quad \left(\forall t_e \in T_e\right) \tag{12}$$

[0060]

Constraint equation (12) indicates that if any one of the candidate tree graphs t_e is used for accommodating a path, r^{te} is set equal to 1.

[0061]

Finally, the optimizing means 105 solves the compound integer programming problem generated by the optimizing threshold generating means 101, the tree defining equation generating means 102, the path accommodating equation generating means 103 and the tree utilization check equation generating means 104 by using the simplex method described in the aforesaid "Optimizing Handbook" to obtain a tree structured communication channel with a minimum number of trees (see step 305, Fig 3). Note that if there is more than one egress node, the present invention is applied to each of

these egress nodes.

[0062]

The following is a description of a second mode of implementation of the present invention. This second mode is implemented by operating the optimizing threshold generating means 101, the tree defining equation generating means 102, the path accommodating equation generating means 103, the tree utilization check equation generating means 104 and the optimizing means 105 of Fig. 1 according to steps 401, 402, 403, 404 and 405 of the flowchart of Fig 4, the difference between the first and the second modes of implementation being the tree defining equation generating means 102 performing step 402 in a different manner from the first embodiment. Therefore, the following description is only concerned with the operation of the tree defining equation generating means 102 according to step 402.

[0063]

Tree defining equation generating means 102 generates constraint equations (step 402, Fig. 4) necessary for the candidate tree graphs to become trees, i.e., necessary to concatenate the candidate tree graphs and further generates constraint equations that are necessary for each ingress node and transit nodes to operate as a source node by using a single link.

[0064]

First, constraint equations for concatenation identical to Equations (2) to (4) of the previous embodiment are generated.

[0065]

Then, the following equations (13) and (14) are generated for the ingress nodes and the transit nodes to use only one link when they act as a source node.

[0066]

$$\sum_{m:(l,m)\in L^{c-c}} h_{(l,m)}^{t_e} = 1 \quad \left(\forall \ell \in N^{edge} \setminus \{e\}, \forall t_e \in T_e\right) \tag{13}$$

[0067]

$$\sum_{m:(l,m)\in L^{c-c}} h_{(l,m)}^{t_e} + o_{(l,e)} h_{(l,e)}^{t_e} = 1 \quad \left(\forall \ell \in N^{core}, \forall t_e \in T_e\right) \tag{14}$$

[0068]

Constraint Equation (13) is associated with the ingress nodes and constraint Equation (14) is associated with the transit nodes. Being identical to constraint Equation (2), Equation (13) can be omitted. In such a case, it is not necessary to consider that transit nodes operate as an outlet node and the following equation can be used.

[0069]

$$\sum_{m:(l,m)\in L^{c-c}} f^{te}_{(l,m)} - \sum_{m:(m,l)\in L^{c-c}} f^{te}_{(m,l)} + o_{(l,e)} f^{te}_{(l,e)}$$

$$- \sum_{m:(m,l)\in L^{e-c}} h^{te}_{(m,l)} = 0 \quad (\forall \ell \in T_e, \ell \in N^{core}) \tag{15}$$

[0070]

The following is a description of a third mode of implementation of the present invention. In this embodiment, trees that are already generated are defined as "existing trees".

[0071]

Fig. 5 is a block diagram of a circuit according to the third embodiment of the present invention for designing a tree structured communication channel. The design circuit of this embodiment comprises an existing path accommodating means 501, an optimizing threshold generating means 502, a tree defining equation generating means 503, a path accommodating means 504, a tree utilization check equation generating means 505, and an optimizing means 506, a computer 500, an input device 507, and an output device 508 including a display unit, and a storage medium 509.

[0072]

Storage medium 509 may be a floppy disk, a read-only memory or any other types of store. Storage medium 509 is used to store a program for enabling the computer 100 to function as a design circuit for designing a tree structured communication channel. This program is read by the computer 500 to control its function so that the existing path accommodating means 501, the optimizing threshold generating means 502, the tree defining equation generating means 503, the path accommodating means 504, the tree utilization check equation generating means 505, and the optimizing means 506 are implemented.

[0073]

Fig. 6 is a flowchart of one example of the operation of the existing path accommodating means 501, the optimizing threshold generating means 502, the tree defining equation generating means 503, the path accommodating means 504, the tree utilization check equation generating means 505, and the optimizing means 506 according to the third embodiment of the present invention.

[0074]

First, there are given, as input data to the computer 500 by using the input device 507, a network topology comprising terminal nodes, transit nodes

and interconnecting links, the node identifier of an egress node, a set of paths between ingress nodes and the egress node, and a set of candidate tree graphs and a set of existing trees.

[0075]

Existing path accommodating means 501 makes a decision whether given paths can be accommodated in existing trees (step 601, Fig. 6).

[0076]

A set of existing trees is defined as T^*_e. There is further defined a constant $j^{te}_{(l,m)}$ which assumes 1 when an existing tree $t_e \in T^*_e$ uses a link (l, m); otherwise 0. The decision as to whether a path $p_{(i,e)}$ can be accommodated in an existing tree t_e is performed according to the following equation (16).

[0077]

$$\sum_{(l,m)\in\left\{L^{p(i,e)}\cap L^{c-c}\right\}} j^{t_e}_{(l,m)} + \sum_{(l,m)\in\left\{L^{p(i,e)}\cap L^{e-c}\right\}} j^{t_e}_{(l,m)} + \sum_{(l,m)\in\left\{L^{p(i,e)}\cap L^{e-c}\right\}} j^{t_e}_{(l,m)} = \left|L^{p(i,e)}\right| \tag{16}$$

[0078

This decision process is performed on all of the existing trees $t_e \in T^*_e$. If one of the decisions meets Equation (16), it means that the path $p_{(i,e)}$ can be accommodated in the existing tree.

[0079]

The decision is performed on all paths whether they can be accommodated in existing trees. If it is determined that all the paths can be accommodated in the existing trees, the routine is terminated. Otherwise, flow proceeds to step where new trees that can accommodate such un-accommodated paths are created (see step 602, Fig. 6).

[0080]

Steps 603 to 607 of Fig. 6 are respectively identical to steps 301 to 305 of Fig. 3 of the first embodiment except that the processed input data are the paths that cannot be accommodated in the existing trees.

[0081]

The following is a description of a fourth mode of implementation of the present invention. As shown in Fig. 5, the design circuit of this embodiment comprises an existing path accommodating means 501, an optimizing threshold generating means 502, a tree defining equation generating means 503, a path accommodating means 504, a tree utilization

check equation generating means 505, and an optimizing means 506. These means are implemented by executing steps 701 to 707 of Fig. 7. Note that the fourth embodiment differs from the third embodiment only with respect to step 704 that implements the tree defining equation generating means 503. Therefore, the description is only concerned with the operation of the tree defining equation generating means 503.

[0082]

Tree defining equation generating means 503 generates constraint equations necessary for the candidate tree graphs to form a tree and constraint equations necessary for the candidate tree graphs to concatenate each other and a constraint equation that each ingress node and the transit node use only one link when then operate as a source node (see step 704, Fig 7). Note that step 704 is identical to step 402 of Fig. 4.

[0083]

The following is a description of a fifth mode of implementation of the present invention.

[0084]

An artificial value is first defined. The artificial variable is a non-negative variable added to the left side of each constraint equation when a linear programming problem is represented by a threshold system. A total sum of artificial variables is minimized by an object function. When a problem in which an object function is set for minimizing the total sum of artificial variables is called a realizability decision problem. If the solution of the realizability decision problem is 0, the problem is one that can be solved by the simplex method (as described in the aforesaid "Optimizing Handbook").

[0085]

Fig. 8 is a block diagram of the design circuit of the fifth embodiment. This design circuit is comprised of a computer 800 including a realizability decision threshold generating means 801, a tree defining equation generating means 802, a path accommodating equation generating means 803, an artificial variable embedding means 804 and an optimizing means 805, an input device 806 including a keyboard, an output device 807 including a display unit and a storage medium 808.

[0086]

Storage medium 808 may be a floppy disk, a read-only memory or any other types of store. Storage medium 808 is used to store a program for enabling the computer 800 to function as a design circuit for designing a tree structured communication channel. This program is read by the computer 800 to control its function so that the realizability decision threshold generating means 801, the tree defining equation generating means 802, the path

accommodating equation generating means 803, an artificial variable embedding means 804 and an optimizing means 805, an input device 806 including the keyboard, an output device 807 including the display unit and the storage medium 808 are implemented.

[0087]

Fig. 9 is a flowchart of the operation of the design circuit of the fifth embodiment. The following is a description of the flowchart of Fig. 9.

[0088]

The following description proceeds in the order of tree defining equation generating means 802, path accommodating equation generating means 803, artificial variable embedding means 804, realizability decision threshold generating means 801, and optimizing means 805.

[0089]

Tree defining means 802 and path accommodating equation generating means 803 operate in the same manner as the tree defining equation generating means 102 and the path accommodating equation generating means 103 of the first embodiment (see steps 902, 903, Fig. 9).

[0090]

Artificial variable embedding means 804 embeds artificial variables into each of the constraint equations (see step 904, Fig. 9). For all of the constraint equations formed by the tree defining equation generating means 802 and the path accommodating equation generating means 803, coefficient matrices a_k, variable vectors x_k, coefficient vectors c_k are established, where k represents the constraint equation. Artificial variable vectors y_k are also defined. If the k-th constraint equation is represented by an equality symbol, the equation in which the artificial variable is embedded is rewritten as Equation (17).

[0091]

$$a_k x + y_k = c_k \qquad (17)$$

[0092]

In this way, the artificial variable embedding means 804 inserts artificial variables into all constraint equations.

[0093]

Realizability decision threshold generating means 801 generates an object function that minimizes the total sum of artificial variables (see step 901, Fig. 9). This is the threshold for making a realizability test. If the object function is 0, it is determined that this problem is executable (realizable).

[0094]

If the result obtained by the optimizing means 805 indicates that the object function is zero, this problem is executable and a tree-structured

communication channel that can accommodate given paths can be obtained (see step 905, Fig 9).

[0095]

If the object function is of a non-zero value, it is determined that a tree-structured communication channel that can accommodate all the given paths cannot be obtained within the range of the given candidate tree graphs.

[0096]

In an alternative embodiment, the artificial variable embedding means 804 may add artificial variables into arbitrary constraint equations, rather than into all the constraint equations.

[0097]

The following is a description of a sixth mode of implementation of the present invention. In this embodiment, the realizability decision threshold generating means 801, the tree defining equation generating means 802, the path accommodating equation generating means 803, the artificial variable embedding means 804 and the optimizing means 805 of Fig. 8 are implemented by executing steps 1001, 1002, 1003, 1004 and 1005 of the flowchart of Fig. 10. This embodiment differs from the fifth embodiment in that step 1002 executed by the tree defining equation generating means 802 differs. Therefore, the description is only concerned with the operation of the tree defining equation generating means 802.

[0098]

Tree defining equation generating means 802 generates constraint equations that enable the candidate tree graphs to become trees (see step 1002, Fig. 10).

[0099]

Next, the following is a description of a seventh mode of implementation of the present invention.

[0100]

Fig. 11 is a block diagram of the design circuit of the seventh embodiment. The design circuit of this embodiment is comprised of a computer 1100 including an existing path accommodating means 1101, a realizability decision threshold generating means 1102, a tree defining equation generating means 1103, a path accommodating equation generating means 1104, an artificial variable embedding means 1105, and an optimizing means 1106, an input device 1107 and an output device 1108 and a storage medium 1109.

[0101]

Storage medium 1109 may be a floppy disk, a read-only memory or any other types of store. Storage medium 1109 is used to store a program for

enabling the computer 1100 to function as a design circuit for designing a tree structured communication channel. This program is read by the computer 1100 to control its function so that the existing path accommodating means 1101, the realizability decision threshold generating means 1102, the tree defining equation generating means 1103, the path accommodating equation generating means 1104, the artificial variable embedding means 1105, and the optimizing means 1106 are implemented.

[0102]

Fig. 12 is a flowchart of the operation of the design circuit of the seventh embodiment.

[0103]

The seventh embodiment is identical to the fifth embodiment with the exception that step for accommodating the given paths into the existing trees and step for making a decision whether all the paths can be accommodated in the existing trees (see steps 1201, 1202, Fig. 12). Steps 1201 and 1202 are identical to steps 601 and 602, respectively, of Fig. 6.

[0104]

The following is a description of an eighth mode of implementation of the present invention. In this embodiment, the existing path accommodating means 1101, the realizability decision threshold generating means 1102, the tree defining equation generating means 1103, the path accommodating equation generating means 1104, the artificial variable embedding means 1105, and the optimizing means 1106 of Fig. 11 are implemented by executing steps 1301, 1302, 1303, 1304, 1305, 1306, and 1307 of the flowchart of Fig 13.

[0105]

The eighth embodiment of the present invention is identical to the seventh embodiment with the exception of step that enables the candidate tree graphs to become a tree (see step 1304, Fig. 13). Further, step 1304 is identical to step 402 of the second embodiment (Fig. 4).

[0106]

[Advantages of the Invention]

The first advantage of the present invention is that it enables paths given between ingress nodes and an egress node to be accommodated in a tree-structured communication channel having a minimum number of tree structures. The reason for this is that a compound integer programming problem having an object function that minimizes the number of tree-structured channels is solved.

[0107]

The second advantage of the present invention is that it enables paths given between ingress nodes and an egress node to be accommodated in a

tree-structured communication channel. The reason for this is that a compound integer programming problem for finding a number of tree-structured communication channels for accommodating paths is solved.

[0108]

The third advantage of the present invention is that, if there is an existing tree-structured channel, it utilizes the existing tree structured channel to reduce the number of newly generated tree-structured channels. The reason for this is that a decision is first made as to whether paths given between ingress nodes and an egress node can be accommodated in the existing tree-structured channel and a new tree-structured communication channel is generated for accommodating those paths which cannot be accommodated in the existing channel.

[BRIEF DESCRIPTION OF THE DRAWINGS]a

[Fig. 1]

A block diagram of a design circuit for designing a tree-structured communication channel according to the first and second modes of implementation of the present invention.

[Fig. 2]

A diagram of a network to which the present invention is incorporated.

[Fig. 3]

A flowchart of the operation of the design circuit of the first embodiment of the present invention.

[Fig. 4]

A flowchart of the operation of the design circuit of the second embodiment of the present invention.

[Fig. 5]

A block diagram of a design circuit for designing a tree-structured communication channel according to the third and fourth modes of implementation of the present invention.

[Fig. 6]

A flowchart of the operation of the design circuit of the third embodiment of the present invention.

[Fig. 7]

A flowchart of the operation of the design circuit of the fourth embodiment of the present invention.

[Fig. 8]

A block diagram of a design circuit for designing a tree-structured communication channel according to the fifth and sixth modes of

implementation of the present invention.

[Fig. 9]

A flowchart of the operation of the design circuit of the fifth embodiment of the present invention.

[Fig. 10]

A flowchart of the operation of the design circuit of the sixth embodiment of the present invention.

[Fig. 11]

A block diagram of a design circuit for designing a tree-structured communication channel according to the seventh and eighth modes of implementation of the present invention.

[Fig. 12]

A flowchart of the operation of the design circuit of the seventh embodiment of the present invention.

[Fig. 13]

A flowchart of the operation of the design circuit of the eighth embodiment of the present invention.

[Explanation of the Threshold Numerals]

100	computer
101	optimizing threshold generating means
102	tree defining equation generating means
103	path accommodating equation generating means
104	tree utilization check equation generating means
105	optimizing means
106	input device
107	output device
108	storage medium
$c_1 \sim c_5$	transit nodes
$e_1 \sim e_{10}$	terminal nodes
500	computer
501	existing path accommodating means
502	optimizing threshold generating means
503	tree defining equation generating means
504	path accommodating equation generating means
505	tree utilization check equation generating means
506	optimizing means
507	input device
508	output device
509	storage medium

800	computer
801	realizability decision threshold generating means
802	tree defining equation generating means
803	path accommodating equation generating means
804	artificial variable embedding means
805	optimizing means
806	input device
807	output device
808	storage medium
1100	computer
1101	existing path accommodating means
1102	reliazability decision threshold generating means
1103	tree defining equation generating means
1104	path accommodating equation generating means
1105	artificial variable embedding means
1106	optimizing means
1107	input device
1108	output device
1109	storage medium

[Document Name] Abstract
[Abstract]

[Object]

In order to generate a minimum number of tree-structured communication channels for accommodating paths given between ingress nodes and an egress node for a communications network in which communication proceeds from an ingress node to an egress node.

[Means for Solution]

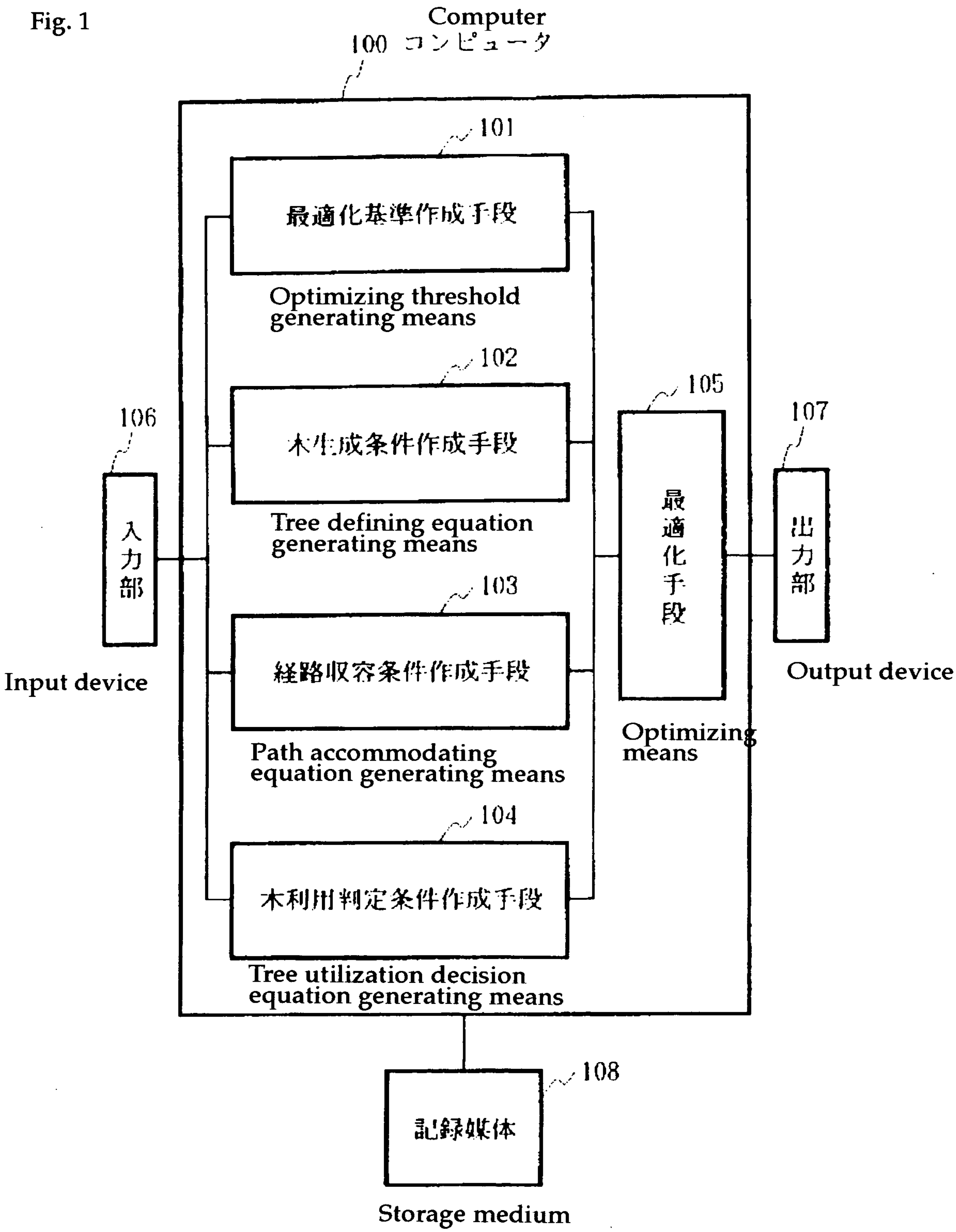
Communication paths are accommodated in a tree-structured communication channel by using an optimizing threshold generating means 101, a tree defining equation generating means 102, a path accommodating equation generating means 103, a tree utilization check equation generating means 104, and minimizing the number of the tree-structured communication channels by solving a compound integer programming problem by an optimizing means 105.

[Selected Drawing] Fig. 1.

【書類名】 図面

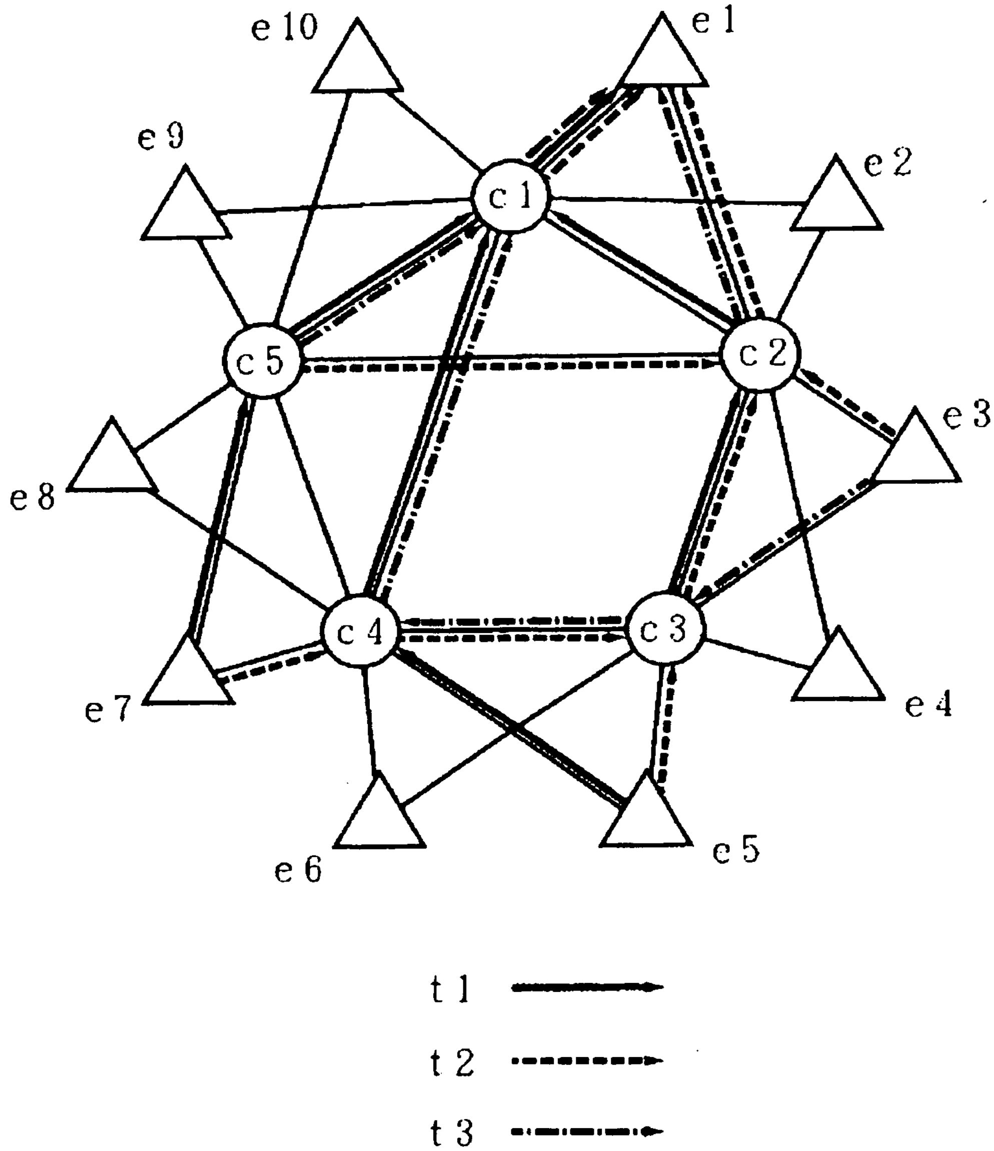
【図１】

Fig. 1



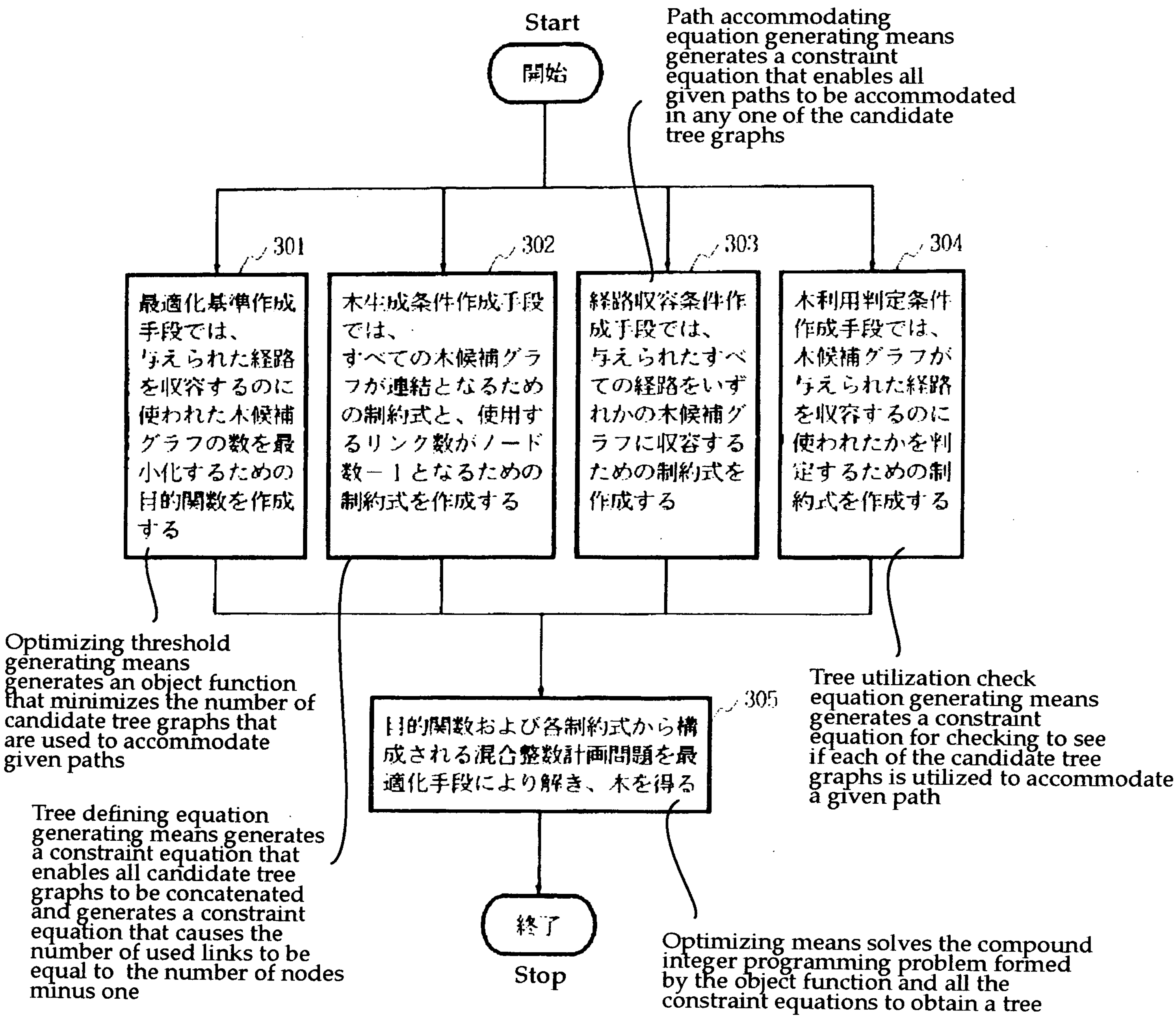
【図２】

Fig. 2



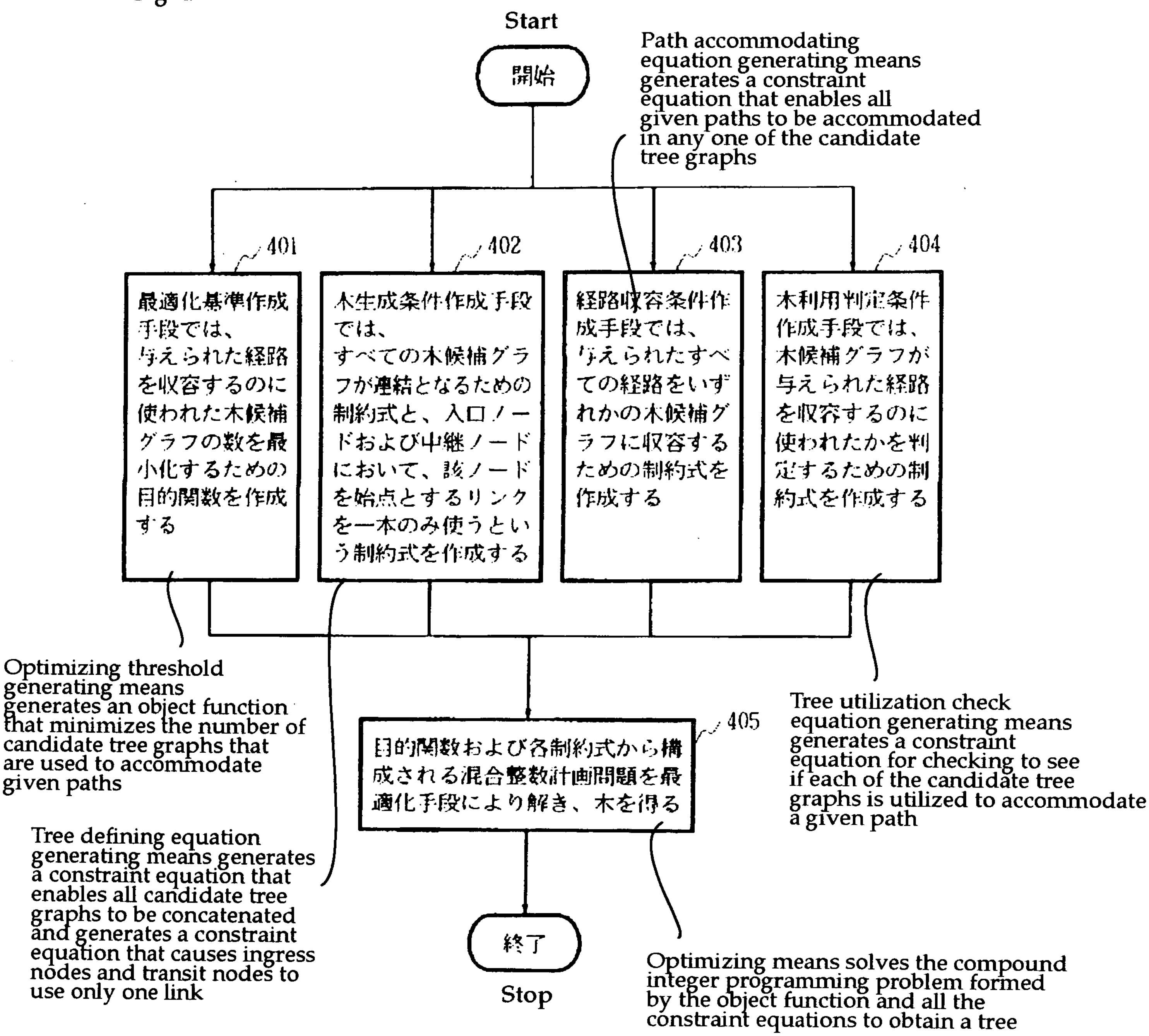
【図3】

Fig. 3

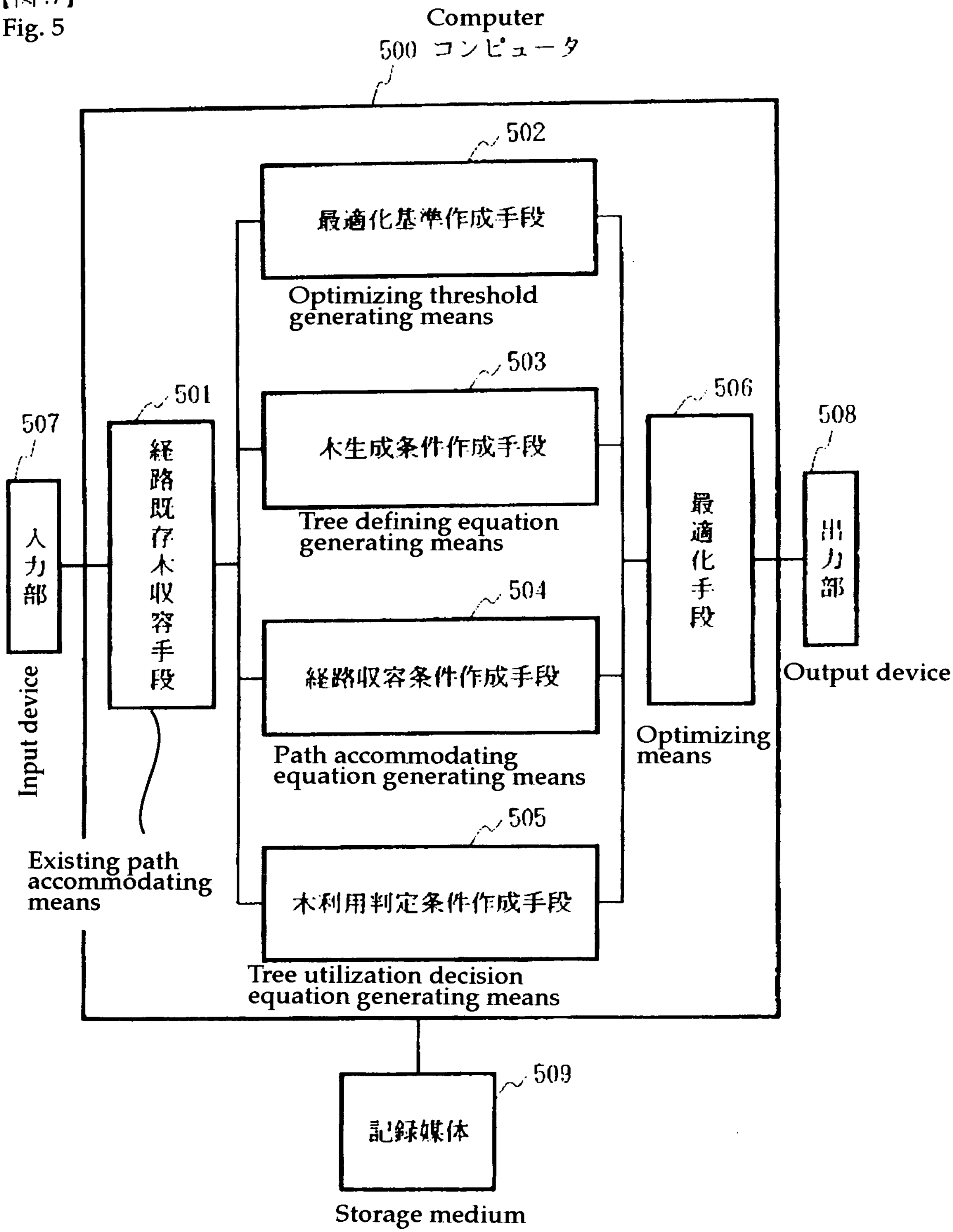

Start
開始
Path accommodating equation generating means generates a constraint equation that enables all given paths to be accommodated in any one of the candidate tree graphs
301
302
303
304
最適化基準作成手段では、与えられた経路を収容するのに使われた木候補グラフの数を最小化するための目的関数を作成する
木生成条件作成手段では、すべての木候補グラフが連結となるための制約式と、使用するリンク数がノード数－1となるための制約式を作成する
経路収容条件作成手段では、与えられたすべての経路をいずれかの木候補グラフに収容するための制約式を作成する
木利用判定条件作成手段では、木候補グラフが与えられた経路を収容するのに使われたかを判定するための制約式を作成する
Optimizing threshold generating means generates an object function that minimizes the number of candidate tree graphs that are used to accommodate given paths
Tree utilization check equation generating means generates a constraint equation for checking to see if each of the candidate tree graphs is utilized to accommodate a given path
305
目的関数および各制約式から構成される混合整数計画問題を最適化手段により解き、木を得る
Tree defining equation generating means generates a constraint equation that enables all candidate tree graphs to be concatenated and generates a constraint equation that causes the number of used links to be equal to the number of nodes minus one
終了
Stop
Optimizing means solves the compound integer programming problem formed by the object function and all the constraint equations to obtain a tree

【図4】

Fig. 4

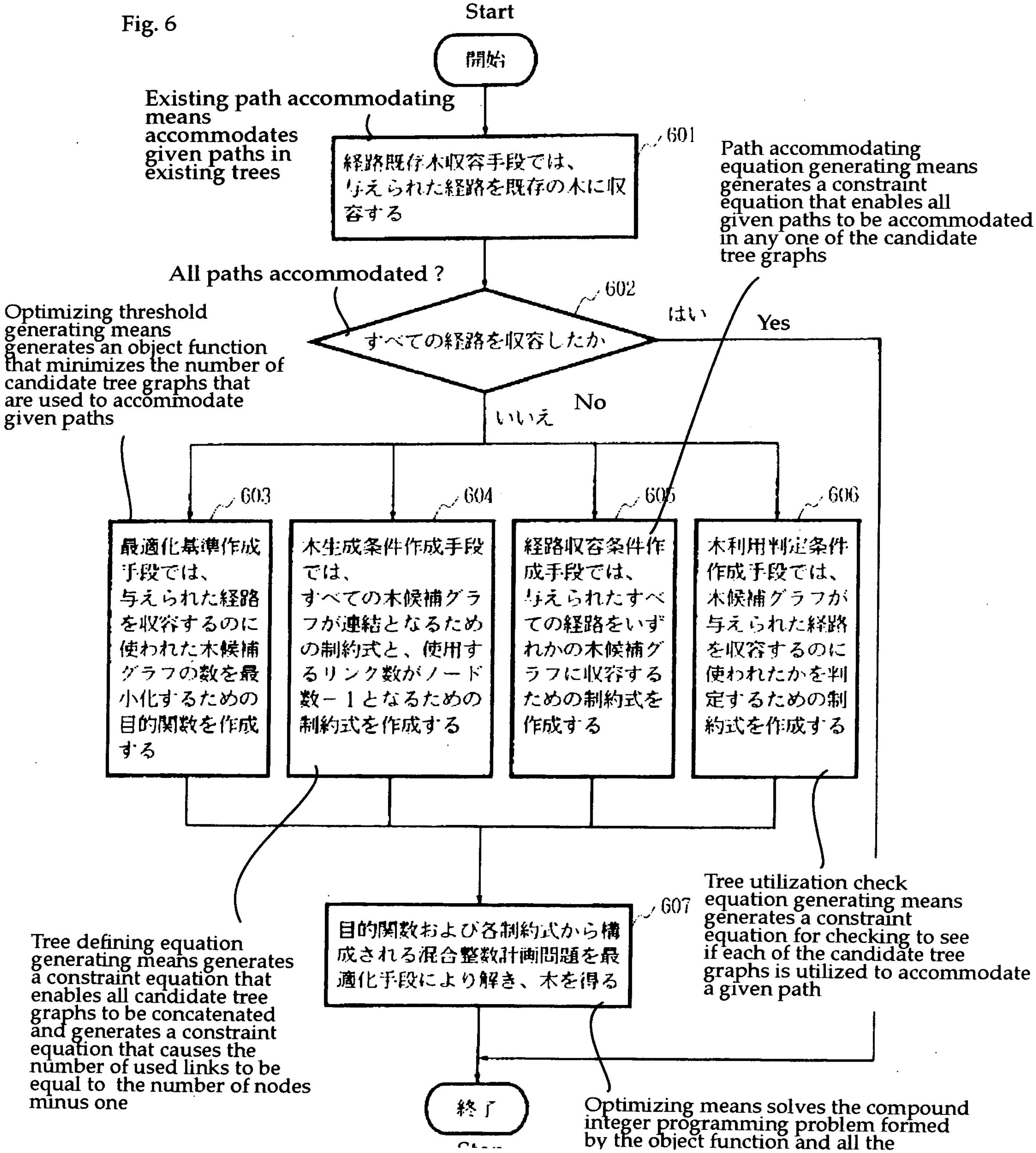

Start
開始
Path accommodating equation generating means generates a constraint equation that enables all given paths to be accommodated in any one of the candidate tree graphs
401
最適化基準作成手段では、与えられた経路を収容するのに使われた木候補グラフの数を最小化するための目的関数を作成する
402
木生成条件作成手段では、すべての木候補グラフが連結となるための制約式と、入口ノードおよび中継ノードにおいて、該ノードを始点とするリンクを一本のみ使うという制約式を作成する
403
経路収容条件作成手段では、与えられたすべての経路をいずれかの木候補グラフに収容するための制約式を作成する
404
木利用判定条件作成手段では、木候補グラフが与えられた経路を収容するのに使われたかを判定するための制約式を作成する
Optimizing threshold generating means generates an object function that minimizes the number of candidate tree graphs that are used to accommodate given paths
Tree utilization check equation generating means generates a constraint equation for checking to see if each of the candidate tree graphs is utilized to accommodate a given path
405
目的関数および各制約式から構成される混合整数計画問題を最適化手段により解き、木を得る
Tree defining equation generating means generates a constraint equation that enables all candidate tree graphs to be concatenated and generates a constraint equation that causes ingress nodes and transit nodes to use only one link
終了
Stop
Optimizing means solves the compound integer programming problem formed by the object function and all the constraint equations to obtain a tree

【図５】
Fig. 5

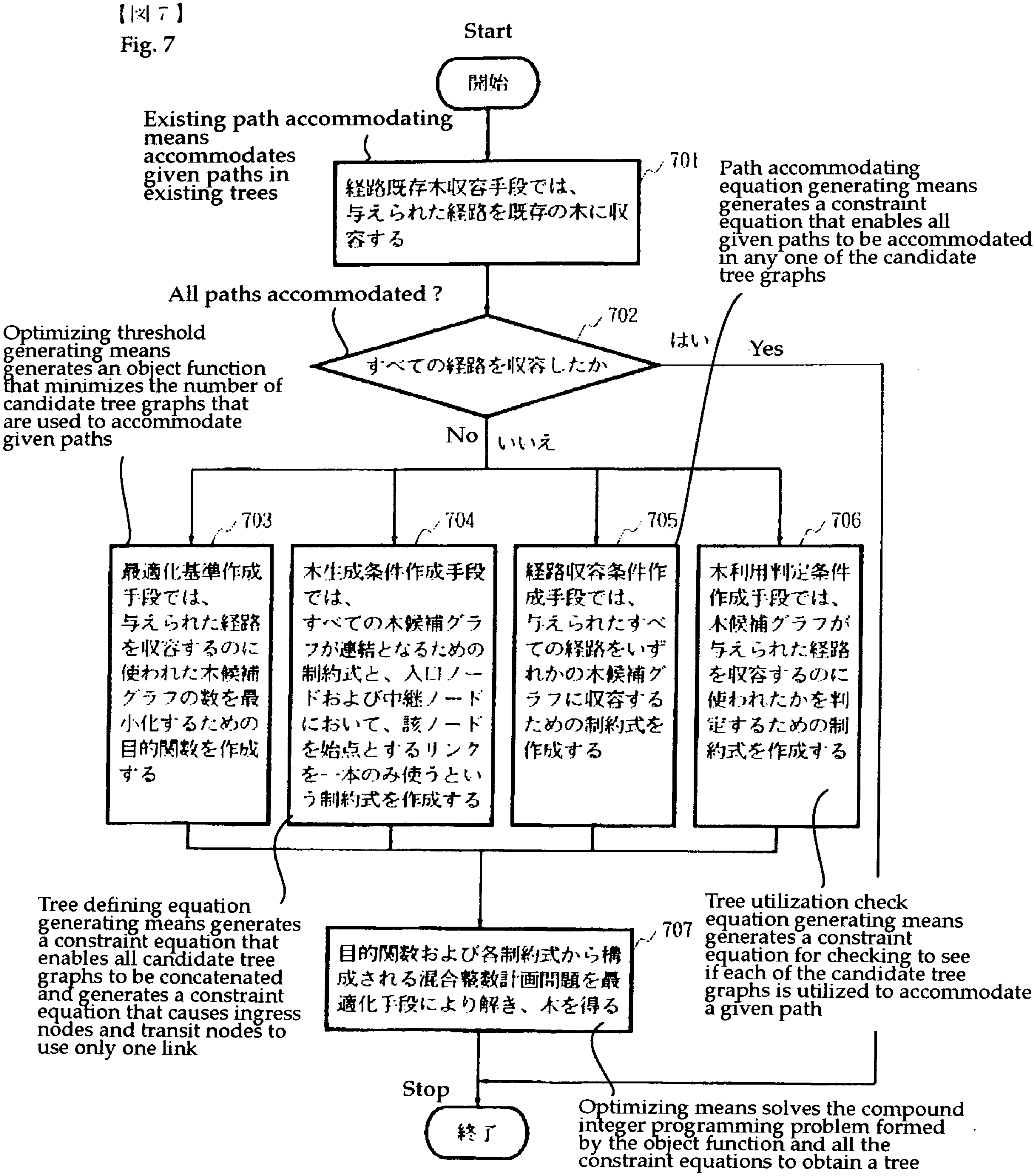

Computer
500 コンピュータ
502
最適化基準作成手段
Optimizing threshold generating means
503
木生成条件作成手段
Tree defining equation generating means
504
経路収容条件作成手段
Path accommodating equation generating means
505
木利用判定条件作成手段
Tree utilization decision equation generating means
501
経路既存木収容手段
Existing path accommodating means
507
入力部
Input device
506
最適化手段
Optimizing means
508
出力部
Output device
509
記録媒体
Storage medium

【図6】

Fig. 6

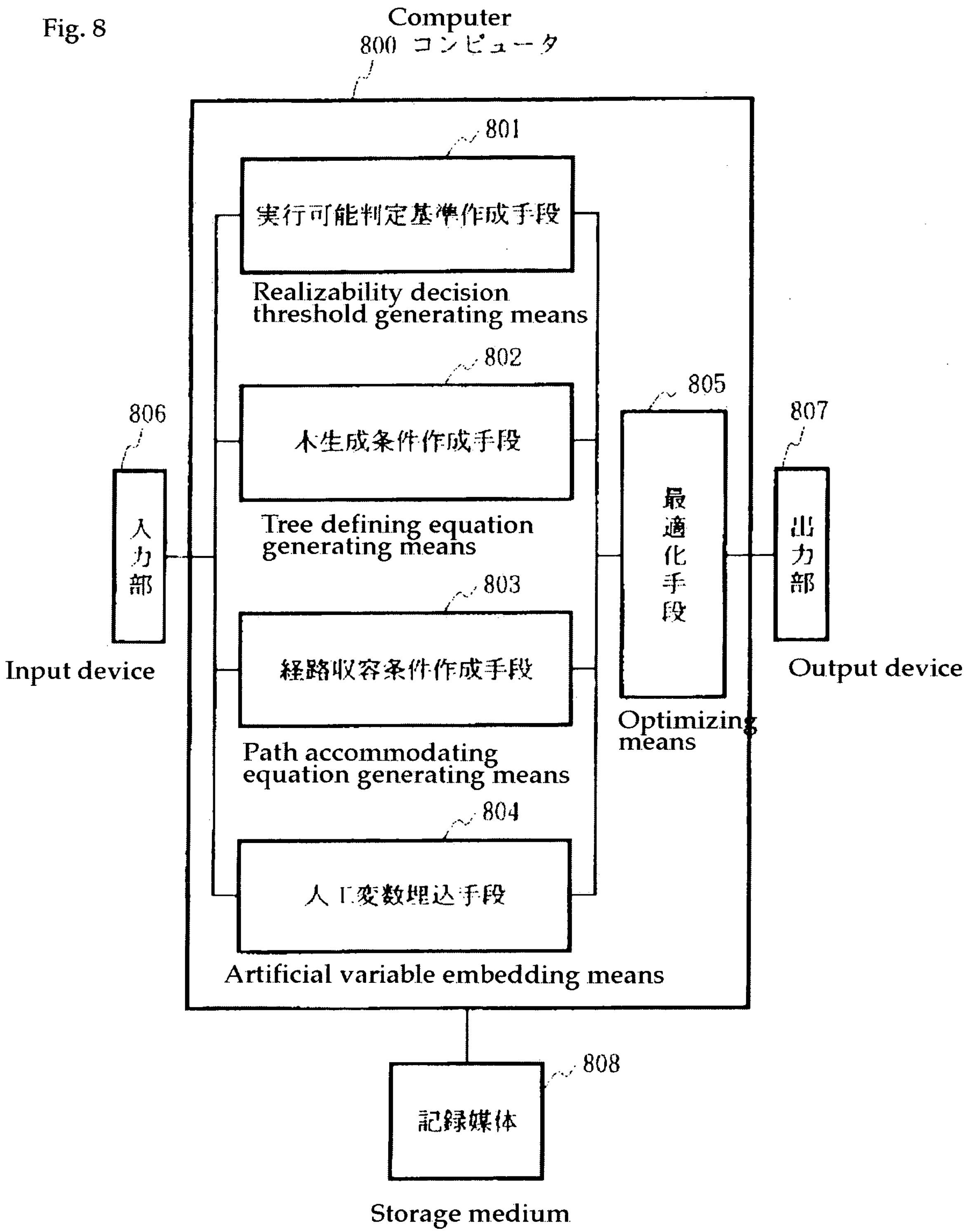

Start
開始
Existing path accommodating means accommodates given paths in existing trees
601
経路既存木収容手段では、与えられた経路を既存の木に収容する
Path accommodating equation generating means generates a constraint equation that enables all given paths to be accommodated in any one of the candidate tree graphs
All paths accommodated ?
602
すべての経路を収容したか
はい
Yes
Optimizing threshold generating means generates an object function that minimizes the number of candidate tree graphs that are used to accommodate given paths
No
いいえ
603
最適化基準作成手段では、与えられた経路を収容するのに使われた木候補グラフの数を最小化するための目的関数を作成する
604
木生成条件作成手段では、すべての木候補グラフが連結となるための制約式と、使用するリンク数がノード数－１となるための制約式を作成する
605
経路収容条件作成手段では、与えられたすべての経路をいずれかの木候補グラフに収容するための制約式を作成する
606
木利用判定条件作成手段では、木候補グラフが与えられた経路を収容するのに使われたかを判定するための制約式を作成する
Tree utilization check equation generating means generates a constraint equation for checking to see if each of the candidate tree graphs is utilized to accommodate a given path
607
目的関数および各制約式から構成される混合整数計画問題を最適化手段により解き、木を得る
Tree defining equation generating means generates a constraint equation that enables all candidate tree graphs to be concatenated and generates a constraint equation that causes the number of used links to be equal to the number of nodes minus one
終了
Optimizing means solves the compound integer programming problem formed by the object function and all the

【図７】
Fig. 7


Start
開始
Existing path accommodating means accommodates given paths in existing trees
経路既存木収容手段では、与えられた経路を既存の木に収容する
701
All paths accommodated ?
すべての経路を収容したか
702
はい
Yes
いいえ
No
Optimizing threshold generating means generates an object function that minimizes the number of candidate tree graphs that are used to accommodate given paths
Path accommodating equation generating means generates a constraint equation that enables all given paths to be accommodated in any one of the candidate tree graphs
703
最適化基準作成手段では、与えられた経路を収容するのに使われた木候補グラフの数を最小化するための目的関数を作成する
704
木生成条件作成手段では、すべての木候補グラフが連結となるための制約式と、入口ノードおよび中継ノードにおいて、該ノードを始点とするリンクを一本のみ使うという制約式を作成する
705
経路収容条件作成手段では、与えられたすべての経路をいずれかの木候補グラフに収容するための制約式を作成する
706
木利用判定条件作成手段では、木候補グラフが与えられた経路を収容するのに使われたかを判定するための制約式を作成する
Tree defining equation generating means generates a constraint equation that enables all candidate tree graphs to be concatenated and generates a constraint equation that causes ingress nodes and transit nodes to use only one link
Tree utilization check equation generating means generates a constraint equation for checking to see if each of the candidate tree graphs is utilized to accommodate a given path
707
目的関数および各制約式から構成される混合整数計画問題を最適化手段により解き、木を得る
Optimizing means solves the compound integer programming problem formed by the object function and all the constraint equations to obtain a tree
Stop
終了

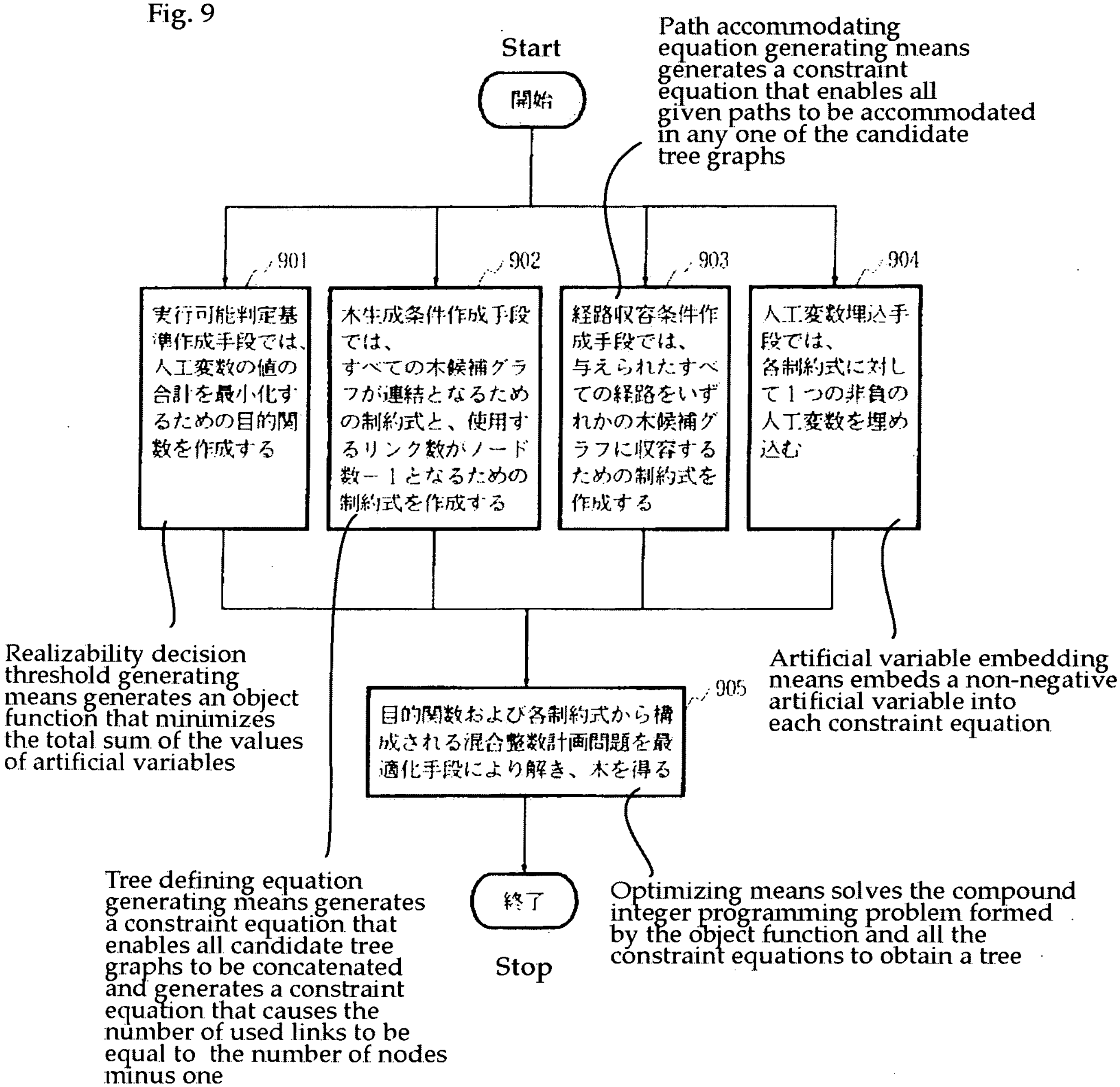
【図８】

Fig. 8

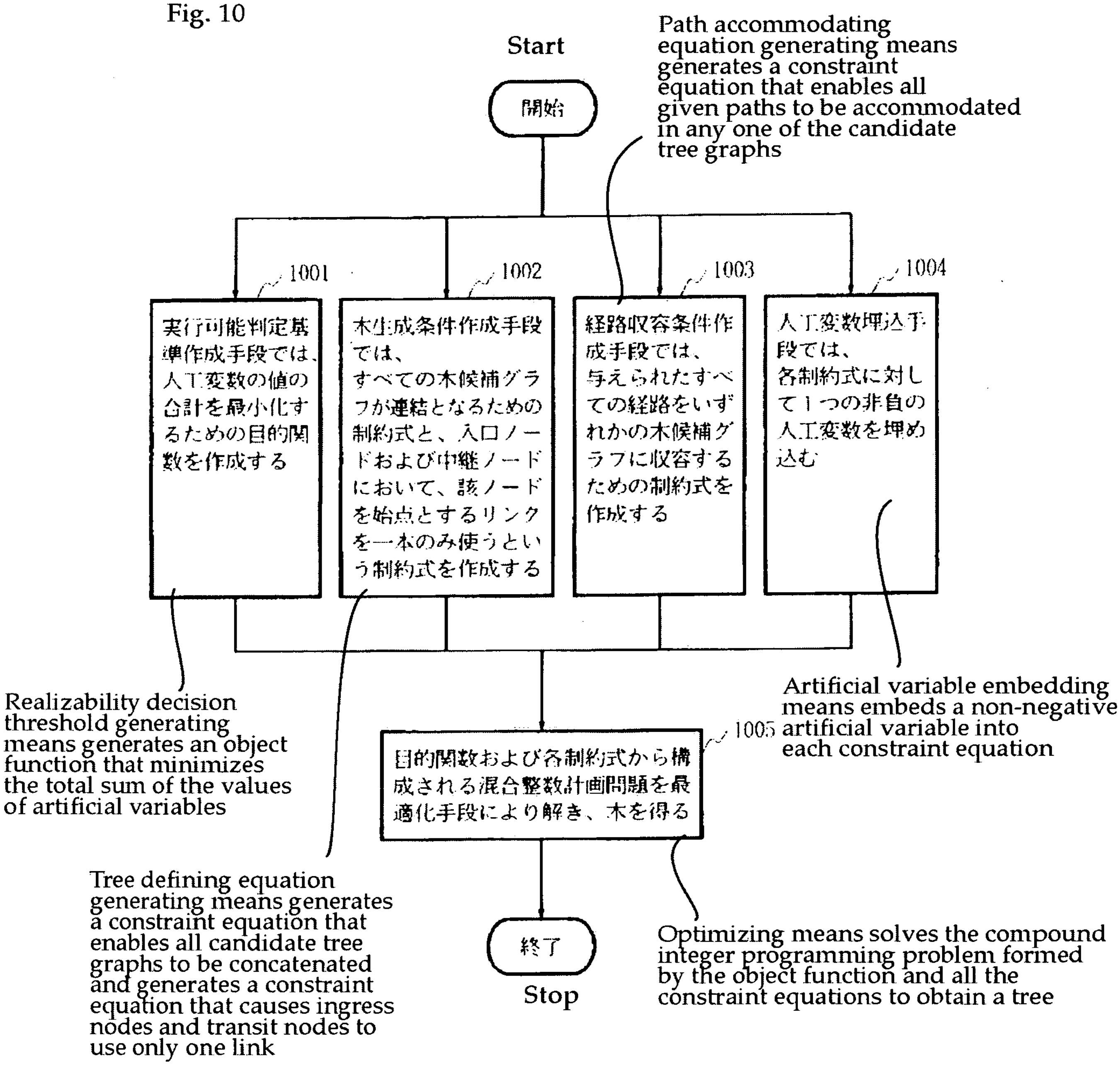

Computer
800 コンピュータ
801
実行可能判定基準作成手段
Realizability decision threshold generating means
802
木生成条件作成手段
Tree defining equation generating means
803
経路収容条件作成手段
Path accommodating equation generating means
804
人工変数埋込手段
Artificial variable embedding means
805
最適化手段
Optimizing means
806
入力部
Input device
807
出力部
Output device
808
記録媒体
Storage medium

【図9】

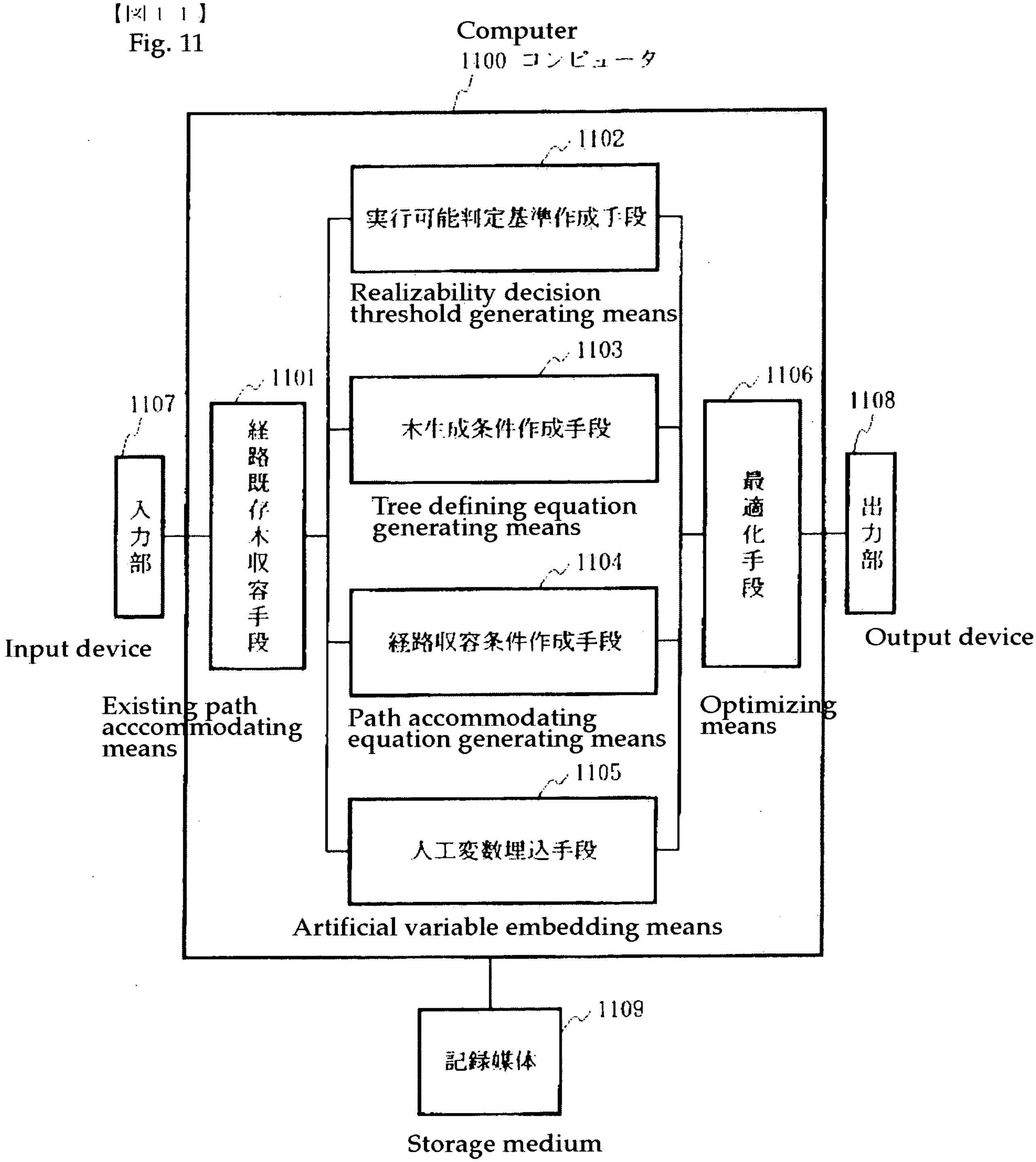
Fig. 9


Start
開始
Path accommodating equation generating means generates a constraint equation that enables all given paths to be accommodated in any one of the candidate tree graphs
901
実行可能判定基準作成手段では、人工変数の値の合計を最小化するための目的関数を作成する
902
木生成条件作成手段では、すべての木候補グラフが連結となるための制約式と、使用するリンク数がノード数－1となるための制約式を作成する
903
経路収容条件作成手段では、与えられたすべての経路をいずれかの木候補グラフに収容するための制約式を作成する
904
人工変数埋込手段では、各制約式に対して1つの非負の人工変数を埋め込む
Realizability decision threshold generating means generates an object function that minimizes the total sum of the values of artificial variables
Artificial variable embedding means embeds a non-negative artificial variable into each constraint equation
905
目的関数および各制約式から構成される混合整数計画問題を最適化手段により解き、木を得る
Tree defining equation generating means generates a constraint equation that enables all candidate tree graphs to be concatenated and generates a constraint equation that causes the number of used links to be equal to the number of nodes minus one
終了
Stop
Optimizing means solves the compound integer programming problem formed by the object function and all the constraint equations to obtain a tree

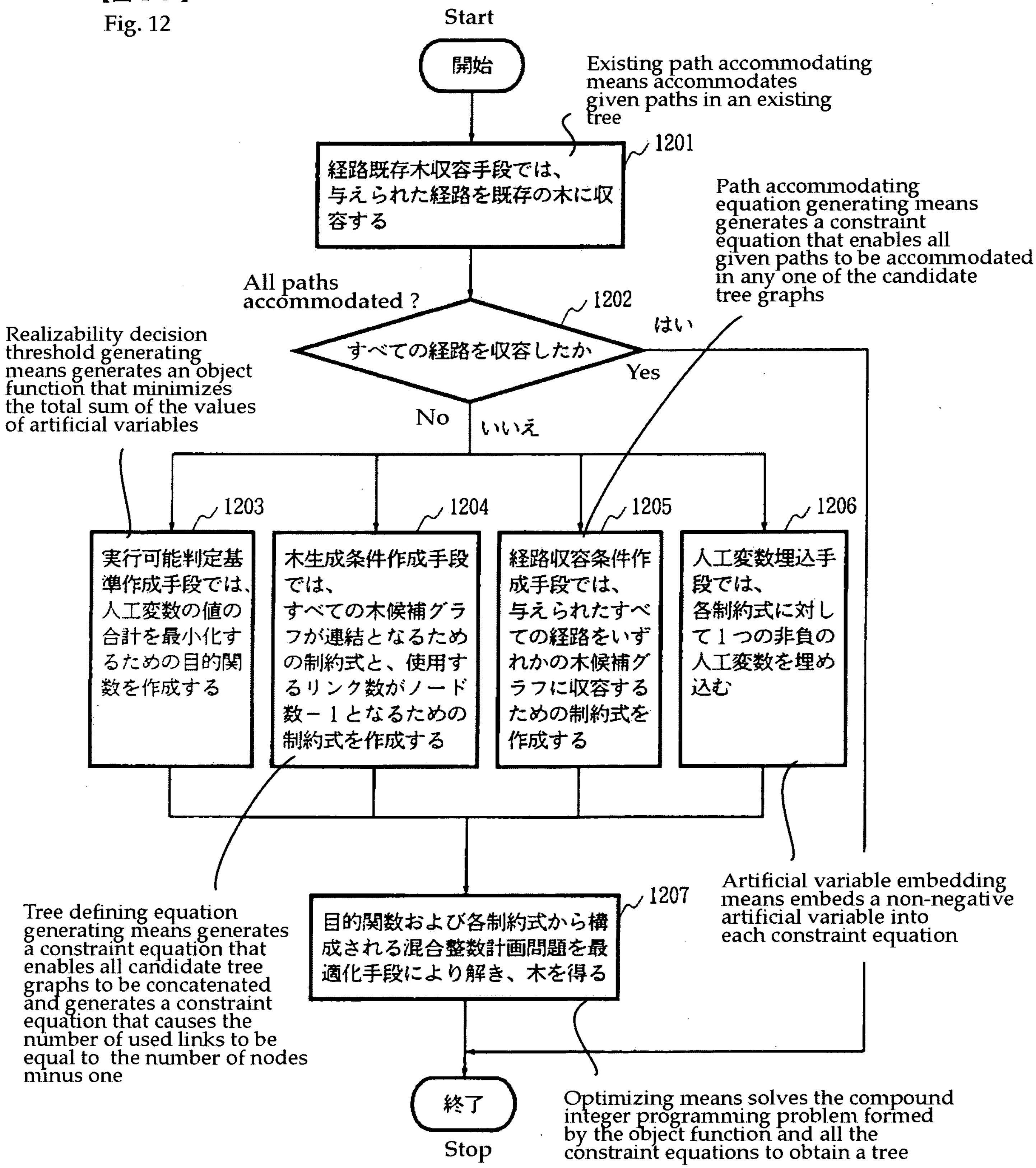
【図１０】
Fig. 10


Start
開始
Path accommodating equation generating means generates a constraint equation that enables all given paths to be accommodated in any one of the candidate tree graphs
1001
1002
1003
1004
実行可能判定基準作成手段では、人工変数の値の合計を最小化するための目的関数を作成する
木生成条件作成手段では、すべての木候補グラフが連結となるための制約式と、入口ノードおよび中継ノードにおいて、該ノードを始点とするリンクを一本のみ使うという制約式を作成する
経路収容条件作成手段では、与えられたすべての経路をいずれかの木候補グラフに収容するための制約式を作成する
人工変数埋込手段では、各制約式に対して１つの非負の人工変数を埋め込む
Artificial variable embedding means embeds a non-negative artificial variable into each constraint equation
Realizability decision threshold generating means generates an object function that minimizes the total sum of the values of artificial variables
1005
目的関数および各制約式から構成される混合整数計画問題を最適化手段により解き、木を得る
Tree defining equation generating means generates a constraint equation that enables all candidate tree graphs to be concatenated and generates a constraint equation that causes ingress nodes and transit nodes to use only one link
終了
Stop
Optimizing means solves the compound integer programming problem formed by the object function and all the constraint equations to obtain a tree

【図11】
Fig. 11

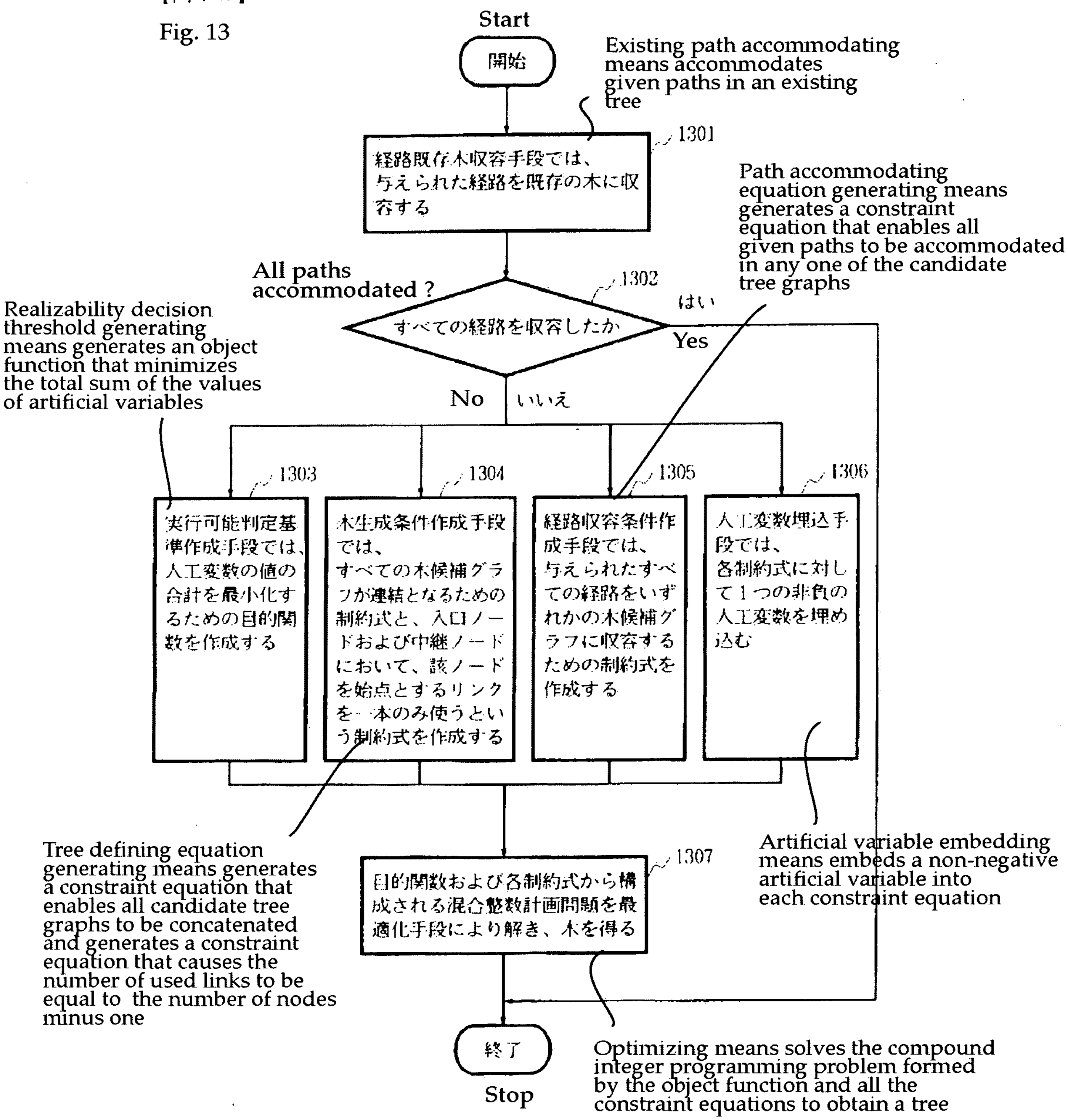

Computer
1100 コンピュータ
1102
実行可能判定基準作成手段
Realizability decision threshold generating means
1103
木生成条件作成手段
Tree defining equation generating means
1104
経路収容条件作成手段
Path accommodating equation generating means
1105
人工変数埋込手段
Artificial variable embedding means
1101
経路既存木収容手段
Existing path acccommodating means
1107
入力部
Input device
1106
最適化手段
Optimizing means
1108
出力部
Output device
1109
記録媒体
Storage medium

【図１２】
Fig. 12


Start
開始
Existing path accommodating means accommodates given paths in an existing tree
1201
経路既存木収容手段では、与えられた経路を既存の木に収容する
All paths accommodated ?
1202
すべての経路を収容したか
はい
Yes
No
いいえ
Path accommodating equation generating means generates a constraint equation that enables all given paths to be accommodated in any one of the candidate tree graphs
Realizability decision threshold generating means generates an object function that minimizes the total sum of the values of artificial variables
1203
実行可能判定基準作成手段では、人工変数の値の合計を最小化するための目的関数を作成する
1204
木生成条件作成手段では、すべての木候補グラフが連結となるための制約式と、使用するリンク数がノード数－１となるための制約式を作成する
1205
経路収容条件作成手段では、与えられたすべての経路をいずれかの木候補グラフに収容するための制約式を作成する
1206
人工変数埋込手段では、各制約式に対して１つの非負の人工変数を埋め込む
Artificial variable embedding means embeds a non-negative artificial variable into each constraint equation
1207
目的関数および各制約式から構成される混合整数計画問題を最適化手段により解き、木を得る
Tree defining equation generating means generates a constraint equation that enables all candidate tree graphs to be concatenated and generates a constraint equation that causes the number of used links to be equal to the number of nodes minus one
終了
Stop
Optimizing means solves the compound integer programming problem formed by the object function and all the constraint equations to obtain a tree

【図１３】

Fig. 13


Start
開始
Existing path accommodating means accommodates given paths in an existing tree
1301
経路既存木収容手段では、与えられた経路を既存の木に収容する
All paths accommodated ?
1302
すべての経路を収容したか
はい
Yes
No
いいえ
Path accommodating equation generating means generates a constraint equation that enables all given paths to be accommodated in any one of the candidate tree graphs
Realizability decision threshold generating means generates an object function that minimizes the total sum of the values of artificial variables
1303
実行可能判定基準作成手段では、人工変数の値の合計を最小化するための目的関数を作成する
1304
木生成条件作成手段では、すべての木候補グラフが連結となるための制約式と、入口ノードおよび中継ノードにおいて、該ノードを始点とするリンクを一本のみ使うという制約式を作成する
1305
経路収容条件作成手段では、与えられたすべての経路をいずれかの木候補グラフに収容するための制約式を作成する
1306
人工変数埋込手段では、各制約式に対して１つの非負の人工変数を埋め込む
Tree defining equation generating means generates a constraint equation that enables all candidate tree graphs to be concatenated and generates a constraint equation that causes the number of used links to be equal to the number of nodes minus one
1307
目的関数および各制約式から構成される混合整数計画問題を最適化手段により解き、木を得る
Artificial variable embedding means embeds a non-negative artificial variable into each constraint equation
終了
Stop
Optimizing means solves the compound integer programming problem formed by the object function and all the constraint equations to obtain a tree